

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

**RL BASED NETWORK DEPLOYMENT AND RESOURCE MANAGEMENT
SOLUTIONS FOR OPPORTUNISTIC WIRELESS ACCESS FOR AERIAL
NETWORKS IN DISASTER AREAS AND SMART CITY APPLICATIONS**

Ph.D. THESIS

Mehmet ARİMAN

Department of Computer Engineering

Computer Engineering Programme

AUGUST 2023

ISTANBUL TECHNICAL UNIVERSITY ★ GRADUATE SCHOOL

**RL BASED NETWORK DEPLOYMENT AND RESOURCE MANAGEMENT
SOLUTIONS FOR OPPORTUNISTIC WIRELESS ACCESS FOR AERIAL
NETWORKS IN DISASTER AREAS AND SMART CITY APPLICATIONS**

Ph.D. THESIS

**Mehmet ARİMAN
(504162503)**

Department of Computer Engineering

Computer Engineering Programme

Thesis Advisor: Prof. Dr. Berk CANBERK

AUGUST 2023

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

**FELAKET ALANLARI VE AKILLI ŞEHİR UYGULAMALARINDA
UÇAN FIRSATÇI KABLOSUZ ERİŞİM AĞLARI İÇİN TAKVİYELİ ÖĞRENME
TABANLI AĞ OLUŞTURMA VE KAYNAK YONETİMİ ÇÖZÜMLERİ**

DOKTORA TEZİ

**Mehmet ARİMAN
(504162503)**

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Tez Danışmanı: Prof. Dr. Berk CANBERK

AĞUSTOS 2023

Mehmet ARİMAN, a Ph.D. student of ITU Graduate School student ID 504162503 successfully defended the thesis entitled “RL BASED NETWORK DEPLOYMENT AND RESOURCE MANAGEMENT SOLUTIONS FOR OPPORTUNISTIC WIRELESS ACCESS FOR AERIAL NETWORKS IN DISASTER AREAS AND SMART CITY APPLICATIONS”, which he/she prepared after fulfilling the requirements specified in the associated legislations, before the jury whose signatures are below.

Thesis Advisor : **Prof. Dr. Berk CANBERK**
Istanbul Technical University

Jury Members : **Assist. Prof. Gökhan SEÇİNTİ**
Istanbul Technical University

Assoc. Prof. Muhammed Ali AYDIN
Istanbul University

Assoc. Prof. Tolga OVATMAN
Istanbul Technical University

Prof. Dr. Fatih ALAGÖZ
Bogazici University

Date of Submission : **26 May 2023**

Date of Defense : **9 August 2023**





To my children, wife and family



FOREWORD

I want to provide my special thanks to my supervisor Prof. Berk CANBERK for his continuous support during my Ph.D. period. He has always been there for me like a family when I needed help. He has incorporated my cycles and motivated me when the time is right. He also encouraged me to give pause when I needed it most without breaking my motivation. I would only be able to come this far with him.

I want to open a parenthesis for my kids to help me find the power to become a role model for them with the motivation of holding up to teach them not to give up. Their sincere love, smile, and kindness helped me stay focused and energetic despite sleepless nights.

I want to thank my father, mother, and sister for always believing in me and lending their hand without any expectations. This pure love kept me going during hard times.

Last but not least, I would like to thank my wife for her belief in my success and for hanging in there with two kids to help me succeed during this story. I want to stand by her when she achieves the same success and be there for her with our twins to empower her for success.

August 2023

Mehmet ARİMAN
(Computer Engineer, M.Sc.)

TABLE OF CONTENTS

	<u>Page</u>
FOREWORD	ix
TABLE OF CONTENTS	xi
ABBREVIATIONS	xiii
SYMBOLS	xv
LIST OF TABLES	xvii
LIST OF FIGURES	xix
SUMMARY	xxi
ÖZET	xxiii
1. INTRODUCTION	1
1.1 Background.....	1
1.1.1 Problem statement.....	1
1.1.2 Initial motivation	2
1.1.3 Uav environment challenges	5
1.2 Purpose of Thesis	6
1.3 Literature Review	8
1.3.1 Sdn scalability	8
1.3.2 Aerial network deployment strategy.....	11
1.3.3 Aerial communication resource assignment	13
1.4 Hypothesis.....	14
1.4.1 Aerial network deployment for disaster areas.....	15
1.4.2 Opportunistic wifi access for smart-city applications.....	15
2. AERIAL NETWORK ARCHITECTURE	17
2.1 Centralized Aerial Network Deployment	17
2.2 Distributed Aerial Network Deployment	19
2.3 Reward Function	21
3. AERIAL NETWORK SIMULATION ENVIRONMENT	23
3.1 WiFi Environment Simulation.....	23
3.2 UAV Mobility	24
3.3 RL Model Simulation.....	25
4. ENERGY-EFFICIENT RL-BASED AERIAL NETWORK DEPLOY- MENT TESTBED FOR DISASTER AREAS	27
4.1 Introduction	27
4.2 Architecture	30
4.2.1 Ground control station	32
4.2.1.1 Ground controller	33
4.2.1.2 Sdn controller	33
4.3 Use Cases.....	38
4.3.1 Handover mechanism	38
4.3.2 New cell deployment	41

4.4 Performance Evaluation	41
4.4.1 Testbed setup	41
4.4.2 Performance results	46
5. OPPORTUNISTIC RL-BASED WIFI ACCESS FOR AERIAL SENSOR NODES IN SMART CITY APPLICATIONS.....	49
5.1 Introduction	49
5.2 System Model	51
5.2.1 Unmanned aerial sensor network	51
5.2.2 Ground wifi network	51
5.3 Opportunistic Reinforcement Learning Model	52
5.3.1 Environment	52
5.3.2 Actions	52
5.3.3 Deep q network	53
5.4 Model Training.....	54
5.4.1 Data requirements	54
5.4.2 Training method	54
5.5 Benchmark	56
5.5.1 Test setup	56
5.5.2 Performance analysis.....	57
6. CONCLUSIONS	61
6.1 Findings of Centralized Aerial Network Deployment.....	61
6.2 Findings of Distributed Aerial Network Deployment	61
6.3 Future Work	62
REFERENCES	63
CURRICULUM VITAE.....	71

ABBREVIATIONS

3D	: Three dimensional
AN	: Aerial network
AP	: Access point
AVENS	: Aerial Vehicle Network Simulator
BS	: Base station
CapEx	: Capital expense
CommEnt	: Communication Entity
COTS	: Commercial off-the-shelf
CSV	: Comma seperated value
DFS	: Dynamic frequency selection
DNN	: Deep neural network
DQN	: Deep-Q-Network
DRL	: Deep reinforcement learning
FAA	: Federal Aviation Administration
FMU	: Flight management unit
GCS	: Ground control station
IL	: Input layer
ILP	: Integer linear programming
IoT	: Internet of things
IP	: Internet protocol
ISM	: Industrial-Scientific-Medical
ITU	: Istanbul Technical University
KPI	: Key performance indicator
LOS	: Line of sight
LP	: Lithium polymer
LRU	: Least recently used
M2M	: machine to machine
MANET	: Mobile ad-hoc network
ML	: Machine learning
MLP	: Multi layer perceptron
NFV	: Network functions virtualization
NN	: Neural network
OF	: OpenFlow
OL	: Output layer
ONOS	: Open network operating system
QoE	: Quality of experience
QoS	: Quality-of-service
RF	: Radio frequency
RL	: Reinforcement learning
SCF	: Small cell forum

SDN	: Software defined networking
SINR	: Signal to interference noise ratio
SP	: Service provider
TCP	: Transmission control protocol
UAV	: Unmanned air vehicle
UASN	: Unmanned aerial sensor network
UE	: User equipment
US	: United States
VANET	: Vehicular ad-hoc network
VLAN	: Virtual local area network
VNI	: Visual networking index
WAP	: Wireless access point
WiFi	: Wireless fidelity
WS	: Wireless subscribers



SYMBOLS

$\mathcal{F}$	: Set of flow tables
$\mathcal{F}_i$	: Flow table of UAV_i
A	: Operational unmanned air vehicle list
C	: Charging unmanned air vehicle list
$\mathcal{U}$	: Set of user equipment
id_i	: The flow sourced from UE_i
N_{UAV}	: Number of UAVs
N_{UE}	: Number of UEs
TG_i	: Traffic generated by UE_i
$\mathcal{L}$	: Size of the flow table
$\mathcal{P}_{\mathcal{R}_i}$	: Packet statistics of rule i in the flow table
$TN(s_t, a_t, \theta')$	: Target Network
θ	: Weight matrice
$PN(s_t, a_t, \theta)$	: Policy Network
a_t	: Action at time t
s_t	: State at time t
$R(.)$	: Reward



LIST OF TABLES

	<u>Page</u>
Table 4.1 : AN Message Types	36
Table 4.2 : Performance Comparison.....	48
Table 5.1 : Environment Data	52





LIST OF FIGURES

	<u>Page</u>
Figure 1.1 : Increase in Average Traffic Demand vs Peak Traffic Demand	3
Figure 1.2 : The Number of Registered Drones [1]	4
Figure 1.3 : The SDN Based AN Architecture.....	6
Figure 2.1 : Centralized AN Architecture.	18
Figure 2.2 : Distributed AN Architecture.	20
Figure 3.1 : OMNET++ data collection process.	24
Figure 4.1 : Median RTT value on Croatia Earthquake on March 22, 2020 [2]	28
Figure 4.2 : Opportunistic Aerial Network Service Model. [3]	30
Figure 4.3 : MAVLink Message Format [4]	32
Figure 4.4 : Network Control Modules in Ground Control Station. [3]	33
Figure 4.5 : Handover Mechanism. [3].....	36
Figure 4.6 : Sample Handover Mechanism Process with Low Battery. [3]	39
Figure 4.7 : New Cell Deployment Scenario in Theory [3]	41
Figure 4.8 : New Cell Deployment Scenario in Action [3]	42
Figure 4.9 : Test-bed setup on the Drone. [3]	44
Figure 4.10 : UE Coverage trend in testbed-based training of ML model. [3]	45
Figure 4.11 : QoS, packet delivery ratio, the trend in testbed-based training of ML model. [3]	45
Figure 4.12 : Reward trend of testbed-based training of ML model. [3].....	47
Figure 4.13 : Power consumption trend in testbed-based training of ML model. [3]	47
Figure 5.1 : System Model. [5]	49
Figure 5.2 : Training results for the given network. [5]	58
Figure 5.3 : Performance comparison of the proposed method. [5]	59



RL BASED NETWORK DEPLOYMENT AND RESOURCE MANAGEMENT SOLUTIONS FOR OPPORTUNISTIC WIRELESS ACCESS FOR AERIAL NETWORKS IN DISASTER AREAS AND SMART CITY APPLICATIONS

SUMMARY

The growth in the mobile communication area changed the data traffic profile. In addition, the requirement for the deployed infrastructure has significantly changed. The available bandwidth and IP transformation in the mobile backend increased the peak traffic requirements, while the mobile nature of the users changed the required infrastructure over time. The commercial availability of unmanned aerial vehicles potentially addresses requirements changes within the infrastructure. However, its three-dimensional nature and operation range limitations due to limited battery introduce new problems.

Topology control is a significant problem for unmanned aerial vehicle networks. The optimization of the network size for coverage is identical to the minimum set-cover problem. The minimum set-cover problem is NP-hard, even without the service-level agreements enforced within the communication networks. The solution sets provided for tailor-made applications prevent the scalability of aerial networks. The tailor-made solutions require the exact development cost for each new application target. Reinforcement learning provides an ideal solution for addressing requirements for multiple applications with a single development effort. The integration cost depends on data availability for training in reinforcement learning-based deployments. To this end, reinforcement learning is integrated into a central software-defined networking-based control entity to demonstrate the deployment cycle of the aerial network. In addition, the solution's effectiveness is proved by comparing the quality of service, coverage, and power consumption results with existing literature.

Furthermore, the application area of the reinforcement learning is extended to wireless channel selection to address the physical resource assignment problem. The development cost of the model has been the availability of the data. The integration of the new application is demonstrated in the simulation tool to measure the cost. In addition, smart-city application for the aerial network in distributed architecture is simulated with this implementation.

Overall, this thesis conducts a survey of the existing literature on the challenges of aerial networks. In addition, the reinforcement learning integration tool is developed in a simulation format. Finally, the disaster area and smart-city applications are implemented to measure the applicability of the hypothesis.

The comparison results revealed that reinforcement learning-based aerial network topology control provides scalable performance for power consumption while satisfying the quality of service and coverage requirements of the network. In addition,

the improvements in the physical resource allocation for opportunistic access on the wireless medium is proved in wireless channel selection deployment for the smart-city application.



FELAKET ALANLARI VE AKILLI ŞEHİR UYGULAMALARINDA UÇAN FIRSATÇI KABLOSUZ ERİŞİM AĞLARI İÇİN TAKVİYELİ ÖĞRENME TABANLI AĞ OLUŞTURMA VE KAYNAK YONETİMİ ÇÖZÜMLERİ

ÖZET

Kablosuz ağ teknolojilerinde son yıllarda gerçekleşen ilerlemeler sayesinde mobil haberleşmenin kullanım oranı üstel olarak artış göstermiştir. Mobil haberleşmenin yaygınlaşması ile sabit alt yapı uygulamalarının kullanıcı ihtiyacını karşılamada yetersiz olduğu gözlenmiştir. Bunun yanında yeni sabit alt yapı yatırımların da kapasite kullanım oranları incelendiğinde asimetrik kullanım oranı göze çarpmaktadır. Asimetrik kullanım oranlarından faydalanarak mobil altyapı oluşturmak ve optimal dağıtımı sağlamak mevcut ihtiyaçları karşılama anlamında doğru yöne atılacak adım olarak görülmektedir. İnsansız hava araçlarının ticari boyutlara indirilmesi ve raflardan alınabilir hale gelmesi ile birlikte mobil altyapı ihtiyacını karşılamak için uygun çözümün potansiyel adayı da ortaya çıkmıştır.

İnsansız hava araçları ile kurulan ağlar mobil uygulamalar için iyi bir çözüm sağlamasına karşın üç boyutlu uzayda konumlandırma, altyapı erişiminde kullanılacak kaynakların fırsatçı olarak edinimi ve kısıtlı batarya kapasitesi nedeniyle çalışma ve operasyon süresi ile mesafesi de adreslenmesi gereken yeni problemler olarak ortaya çıkmaktadır.

Bunun yanında tek bir uygulama alanı için gerçekleştirilen çözüm içerisinde gereken uzmanlık, mühendislik yatırımı ve geliştirme süresi göz önüne alındığında; eklenecek her uygulama için bu yatırımın gerekmesi önerilecek çözümlerin uygulanabilirliğini yok etmektedir.

Yapay zeka temelli çözümlerin insansız hava araçları ağları için uygulama alanlarının genişletilmesi yeni uygulama eklentilerinin hızlı şekilde oluşturulmasını sağlamak için potansiyeli en yüksek çözümdür. Yapay zeka tabanlı çözümlerde; yeni uygulama eklemek için gerekli olan kısıt, geliştirme zamanı yerine veri toplama ve toplanan veri ile öğrenme döngüsünün çalıştırılması işlemine dönüştürülmektedir.

Bunun yanında sahaya mobil altyapı olarak dağıtılan insansız hava aracı ağlarının merkezi ve dağıtık kontrolü mümkün olmaktadır. Merkezi kontrol ile birlikte optimal karar verme süreci kolaylaşmasına karşın; kontrol elemanı ile sahadaki insansız hava araçları arasındaki iletişim ihtiyacı operasyon alanını kısıtlamakta ve karar iletim döngüsünde gecikmelere neden olmaktadır. Bunun yanında dağıtık mimarideki çözümlerde ağın karar ve uygulama hızı artarken karar kalitesini artırmak için geliştirilmesi gereken modelin ve sağlanması gereken verinin maliyeti artmaktadır.

Tez kapsamında yapay zeka temelli çözümlerin insansız hava aracı ağlarında kullanılabilirliği ve uygulama sonuçlarının ölçülebilmesi amacı ile Python tabanlı bir simülasyon aracı geliştirilmiştir.

Merkezi kontrol ile insansız hava aracı altyapısı dağıtımının deprem gibi felaket senaryolarında uygulanması için gerekli olan mimarinin tasarımı gerçekleştirilmiştir. Bu tasarım içerisinde servis kalitesi, kapsama ve güç tüketimi parametrelerinin önceden belirlenmiş servis seviyesi anlaşmalardaki kısıtlamaları sağlaması hedeflenmiştir. Güç tüketim parametresi ağın ölçeklenebilirliği için kısıt olarak koyulmuştur. Bu çalışma çerçevesinde ağda yer alan insansız hava araçlarının sayısını yöneten bir yapay zeka modeli gerçekleştirilmiştir. Yapay zeka modeli insansız hava aracının konumu, batarya bilgisi, akis tablosu yoğunluğu ve kapsadığı kullanıcı bilgilerini; insansız hava aracının komsusuna ait bilgiler ile harmanlayarak yeni zaman aralığında insansız hava aracının aksiyonu için kararı oluşturmaktadır. Karar aşamasında; mevcut durumu koru, şarj için geri don ya da sahaya çık kararları verilmektedir. Gerçekleştirilen çözüm içerisinde kullanıcılar ile insansız hava aracı arası altyapısız rastgele erişimli kablolu ağ; yer komuta merkezi ile insansız hava aracı arasında ise 433Mhz geniş bant haberleşme protokolleri kullanılmıştır. Çalışma sonucunda güç açısından ölçeklenebilir ve insansız hava aracı ağlarında yeni uygulamalar için taşınması kolay yapay zeka modeli ve bunun simülasyon ortamı ve test düzeneği ile deneyimlenmesi tamamlanmıştır.

Yapay zeka uygulamalarının insansız hava aracı algarına uygulanması sonrası taşınabilirliğini deneyimlemek amacı ile akıllı şehir uygulamalarında kullanım senaryosu tasarımı gerçekleştirilmiştir. Akıllı şehir uygulamalarında mobil sensor düğümü olarak kullanılan insansız hava araçlarının mevcut kablolu ağ kanallarındaki fırsatlardan faydalanarak buluta veri gönderdiği senaryo gerçekleştirilmiştir. Bu senaryo içerisinde kablolu ağlarda fırsat pencerelerine ait boş kanalların tespiti ve veri iletimi için kullanılmasını sağlamak için yapay zeka modelinden faydalanılmıştır. Geliştirilen yapay zeka modelinin eğitimi için açık kaynak veri tabanlarından faydalanılarak kablolu ağ için konum, zaman ve tüm kanallara ait alınan sinyal gücünün gürültüye oranı verisi edinilmiştir. Bu veri kullanılarak felaket senaryolarında topoloji kontrolü için kullanılan çok katmanlı almaç mimarisindeki yapay zeka modeli kanal tahmini için devşirilerek yeniden eğitilmiştir. Eğitilen model ile sensörden elde edilen verilerin iletim oranı ölçülerek modelin kanal tahminindeki başarımları edinilmiştir. Bu başarımların değeri piyasada mevcut olan kablolu erişim noktalarında gerçekleştirilen algoritmalar ile karşılaştırılarak, bu sabit modellere karşı elde ettiği veri ve öğrenim kuvveti sayesinde ilerleme sağladığı gözlemlenmiştir.

Tez kapsamında gerçekleştirilen felaket senaryosu ve akıllı şehir uygulamaları; polinom zamanda çözülemeyen problemlerin yapay zeka yardımı ile eğitim ve veri maliyetine katlanılarak sabit zaman içerisinde yakınsayan çözümlere kavuştuğunu ispatlanmıştır. Bunun yanında yeni uygulama dağıtım maliyetinin veriye erişim ve eğitim olduğu gösterilmiştir.

Çalışmanın gerçekleştirilmesi aşamasında gerçekleştirilen literatür taramasında önerilen çözümlerin bir çoğunun gerçeğe dayanmayan varsayımlar nedeni ile gerçekleştirilemediği tespit edilmiştir. Bu nedenle felaket senaryosu mimarisi için gerçek bir insansız hava aracı ağı için test sistemi tasarlanmış ve saha da test edilmiştir.

Bunun yanında yapay zeka ile gerçekleştirilen modellerin gerçekleştirilmesi için gerekli eğitim verisine erişimin kısıtlılığı gözlemlenmiş ve ileride yapılacak is listesine; mevcut geliştirilmiş olan insansız hava aracı ağı gerçekleştirilmesi üzerinden uzun süreli kampus verisi toplanması ve araştırma için açık kaynak olarak paylaşılması önerilmiştir.

Merkezi kontrol mimarisi içerisinde kontrolcü bağlantısına duyulan ihtiyacın insansız hava aracı ağlarının operasyon alanını kısıtladığı tespit edilmiş ve akıllı şehir modelinde deneyimlenen dağıtık karar mimarisi ile gerçekleştirilen uygulamaların felaket senaryolarına uyumlandırılması için çalışmaların gerçekleştirilmesi önerilmiştir.





1. INTRODUCTION

1.1 Background

1.1.1 Problem statement

The use of unmanned air vehicles (UAVs) become widespread with the reduced cost and increased accessibility of commercial equipment. The underlying factors which trigger this trend are explained in [6]. Furthermore, Figure 1.2 depicts the cumulative increase trend in registered UAVs in the United States (US) since 2015 as reported by Federal Aviation Administration (FAA) [1]. Moreover, the target application of UAVs has also been diversified to a broad range of applications. These applications include but are not limited to video and photo shoots, flying base stations, environmental sensing, border security, Search & Rescue, etc.

Furthermore, all of these examples require an interaction with UAVs to satisfy the purpose of the target application. The sample purpose of interactions is sending commands and receiving mission and application data. The communication interface of the UAV defines the limitations and user experience of the target applications. This thesis mainly focuses on reusing existing technologies to facilitate wireless communication for UAVs so that its use case is easily broadened using commercial off-the-shelf (COTS) devices. To this end, Wireless Fidelity (WiFi) is selected as the primary communication interface for the UAV-based applications inspected within the context of this thesis. The ad-hoc nature of the UAV networks correlates and creates good chemistry with the already supported ad-hoc mode of WiFi protocol.

Furthermore, this thesis narrows the target applications to two different use cases. The first is aerial network support in disaster areas to maintain and focus the S&R operations. The second example is the ever-growing smart city, especially delay-tolerant city surveillance applications.

1.1.2 Initial motivation

The agile and ever-changing nature of ad-hoc network deployments such as UAV networks generates a complex interconnection problem for the underlying network's backhaul communications and physical resources management. The global view of the Software Defined Networking (SDN) solutions increases the efficiency of the temporary infrastructure and physical resource selection problems by exploiting SDN's global view of the current network state. There have been numerous research on the SDN paradigm during the last 10-year period. The benefits of SDN deployments are well-understood and applied to various use cases. SDN-based solutions apply an instant feedback mechanism by continuously monitoring global environmental changes.

Furthermore, SDN enables fine-grained service orchestration by installing flow rules on edge devices. Though the edge devices act upon the upper layers of the Transmission Control Protocol (TCP) / Internet Protocol (IP) stack, the complexity of the edge devices is reduced drastically. The complexity reduction is due to the simple nature of the Open Flow (OF) switches as described in [7]. The OF switch only matches and applies rules within its OF table. The content of the OF table is filled by the powerful servers placed on the northbound of the SDN architecture, usually within the cloud.

The behavior of mobile data usage is evolved with the introduction of next-generation wireless networks combined with cloud services such as video streaming and social media applications. Moreover, these users' quality of Service (QoS) expectations are also escalated. Furthermore, the demand for data is increasing. Though the growth in overall data demand seems to follow a linear trend, change in user behavior causes exponential growth in traffic within peak hours as stated in [8]. These trends are compared in [8] and depicted in Figure 1.1. Cisco Visual Networking Index (VNI) [8] explains the rationale behind this exponential growth by defining the characteristics of broadly used technologies: video and streaming traffic.

While next-generation mobile network infrastructure is being developed, there have been several efforts to address the asymmetric increase described above. The widely

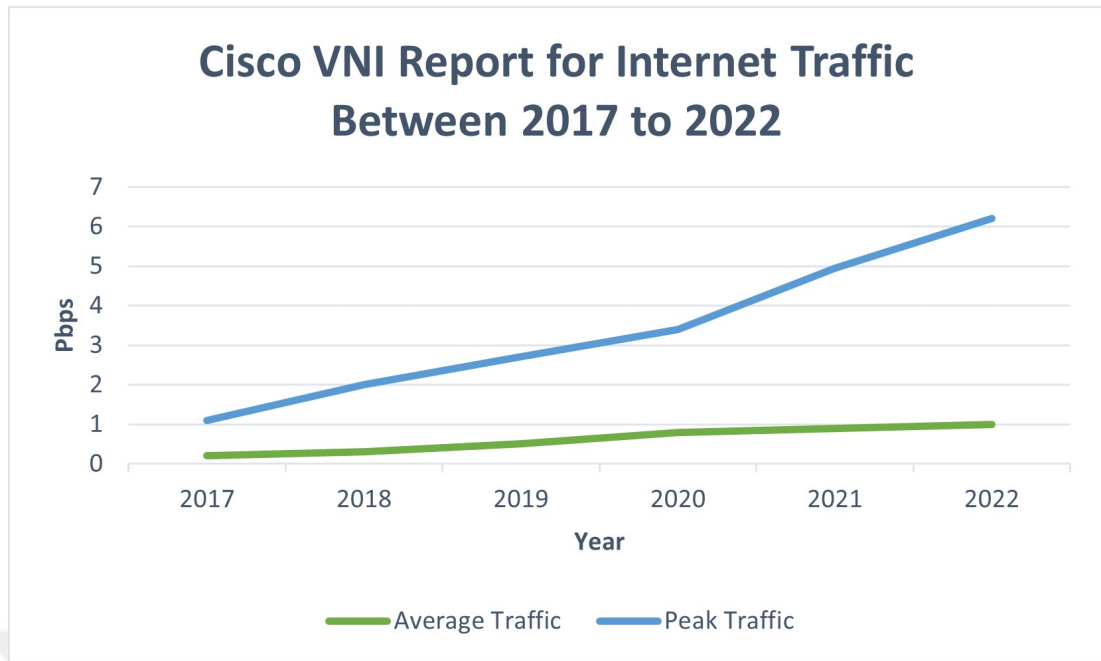


Figure 1.1 : Increase in Average Traffic Demand vs Peak Traffic Demand

accepted proposed solution to this issue is the dense deployment of small cells. In addition, a small cell forum (SCF) is held to present and discuss the reports regarding the feasibility of the proposed approach. Though dense cellular network deployments have the potential to resolve the problems mentioned earlier, this solution introduces new challenges for Service Providers (SP). Coverage area optimization and waste of capital as these new cells will be under-utilized except for the peak hours are two of the major concerns that need to be addressed for feasible deployments, as mentioned in [9]. In [9], deployment sites are picked using a strategy generated by defining a trade-off between user satisfaction and capital expense (CapEx). However, a dense deployment strategy must provide a global solution for mass deployments.

The cell underutilization during low hours prevents cost-optimal deployments for static dense deployment strategies. The increase in utilization is possible by moving under-utilized small cells between sites with different characteristics in peak hours. To this end, there is a serious effort put in both academy and industry to define a solution candidate for this purpose. Though the idea is top-notch, novel works prepared mainly aim to use UAVs to transport small cells. The commercial UAV industry's growth and equipment availability hold the foundations for using UAVs for transportation.

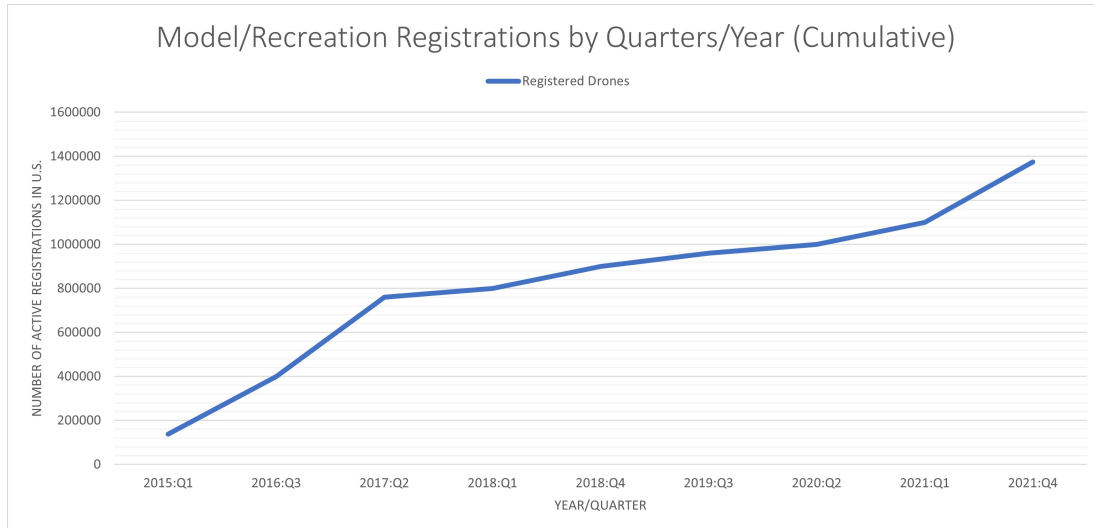


Figure 1.2 : The Number of Registered Drones [1]

Commercial implementations beyond the academy and industry have accelerated the pace to reduce the cost of UAV devices. The commercialization trend of UAVs is explained in [6] and depicted in Figure 1.2. Though the commercial appliance of UAVs gained feasibility over time, the orchestration of small-cell transfer and optimization of small-cells transferred to the target site remains a challenge for feasible deployments. There has been numerous research on SDN paradigm during the last 10-year period. The benefits of this solution are well-understood and applied to different areas. A global view of the network due to centralized controller deployment enables the paradigm to generate an optimal solution for the current state of the network. SDN-based solutions could measure density changes in terms of flow and user within the network by exploiting its global view. This feature makes SDN the best actor for the decision-making process for UAV-based small-cell migration throughout the network. SDN also enables fine-grained service orchestration and deployment with its novel architecture where end devices only apply rules, Open Flow (OF) Tables, that are placed on their flow tables by a centralized entity, SDN Controller. SDN end devices are named OpenFlow Switches as described in [7]. OF Switches require fast-lookup operation for applying flow rules installed by SDN Controller. TCAM memories in hardware provide fast-lookup support. However, the space complexity and power consumption of TCAM memory limit its availability. Moreover, the scalability of

SDN-based networks depends on the optimal use of TCAM memories placed on OF Switches.

1.1.3 Uav environment challenges

At first glance, the motivation is to attack the resolution of limited TCAM resources to offload traffic using aerial base stations carried by UAVs. The idea is to classify the flows for placing tactfully within the limited TCAM memory. Better utilization of TCAM resources tends to increase the processing capability of SDN edge devices. However, the idea of using static classification for the flows needs to be revised to address the aerial networks' agile environment. Static classification limits the framework to tailor-made use cases. To this end, a literature survey on existing solutions for dynamic flow classification proved that reinforcement learning (RL) is one of the most suitable candidates with its application areas of classifying unlabeled data with experience replay.

The aerial network's UAV deployment and resource management infrastructure lies at the heart of the problem. To this end, the core of the problem, UAV network configuration, and physical resource management, moved to the primary focus of this thesis. First, UAV regulations published by aviation authorities like FAA are studied to assess the effect of regulations on UAV utilization. The study's primary finding showed that the regulatory-obedient cost-effective solutions must catch up regarding flight perimeter. Then, a literature survey on the communication link between ground control station (GCS) and UAV proved that physical resource allocation is vital to improving the bandwidth and robustness of the interaction. To this end, aerial network deployment strategy, backend and access network infrastructure, and physical resource assignment problems in ad-hoc aerial WiFi networks are moved to the primary focus of this thesis. RL approaches to support network deployment strategy are visited. Then the RL approach is extended for distributed deployment strategy to support physical resource assignment and WiFi channel selection.

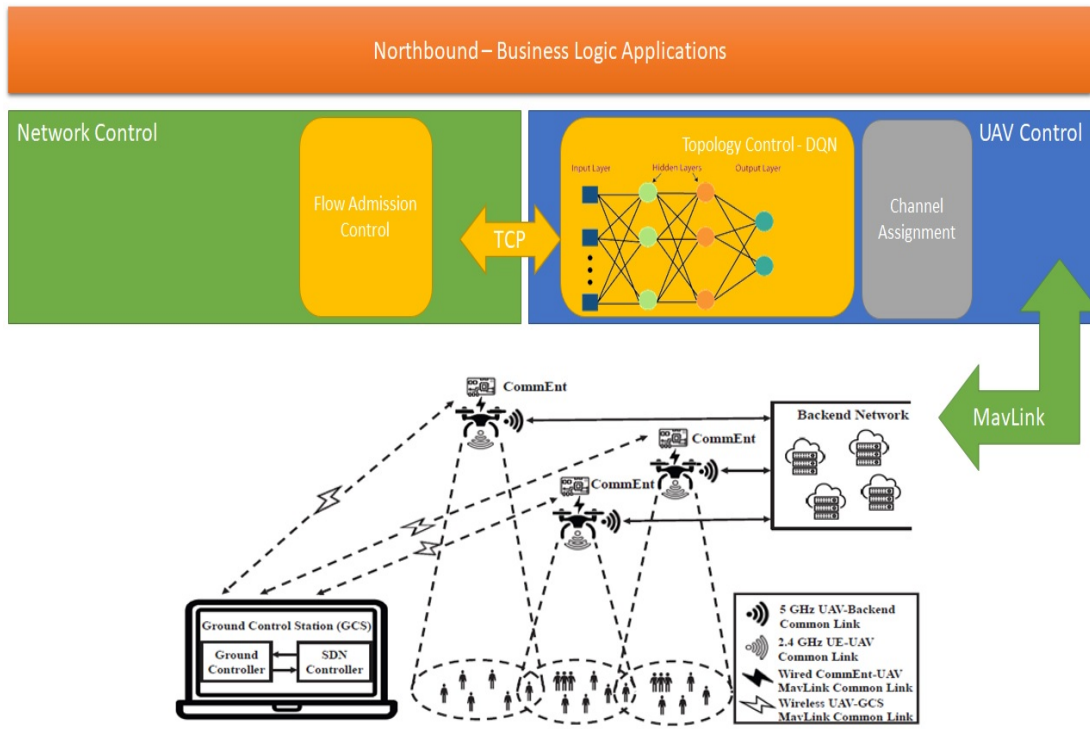


Figure 1.3 : The SDN Based AN Architecture.

1.2 Purpose of Thesis

The section 1.1 provides essential information about the background of the existing problems. The detailed review of the literature in section 1.3 highlights the open research areas in ANs. The study of background in section 1.1 created the architecture given in figure 1.3. Though the northbound applications of the SDN are equally essential and studied in the target applications, the core of the thesis focus on the aerial network architecture and communication infrastructure. This thesis targets scalability and access network resource limitations of the ANs.

Scalability has two significant perspectives in ANs. The first perspective is energy consumption. The thesis aims to optimize the average per-bit power consumption to scale the AN deployments in large deployments. The second aspect is the deployment method. The deployment strategy of the AN could be more efficient and requires significant effort to retrieve data to deploy UAVs to new applications. In addition to data retrieval, the architectural model creation also brings additional development,

setup, and customization costs, preventing the spread of UAV deployments to new applications.

The use of RL in AN applications has the potential to ease its deployments for new areas. The cost of new deployment in RL-based solutions is the training. The cost of development and setup is negligible once the infrastructure works for a single environment. The data collection, alignment, augmentation, and filtering process is not easy, but the big-data era with next-generation communication networks also eases access to the data.

The existing work on RL for AN focuses on the path trajectory rather than addressing network resource limitations. In addition, there needs to be more recent work to integrate RL using the AN deployment strategy. Most of the existing work depends on ILP and game theory. ILP and game theory-based solutions are tailor-made for specific applications and are not easy to reuse for new deployments.

There needs to be more effort in the literature to deploy proposed work into testbeds to measure the assumptions' accuracy. In addition, the available accurate data for training AN deployments also depends on the availability of the testbeds for the researchers.

The operation of the UAV network is sensitive and highly regulated due to security issues. In addition, the availability of the infrastructure equipment and recovery time in case of hazard is significant. That is why developing simulation tools for assessing the optimality of the proposed solutions is essential for developing ANs.

To this end, the purpose of the thesis is:

- Provide background information on ANs use cases, existing problems, and existing work in the literature
- Integrate RL-based solutions for reducing the cost of deployment for new ideas and use cases to scalability and resource assignment problems
- Develop a simulation tool to ease the evaluation of the network deployment and resource assignment problems

- Develop a testbed to ease data collection and provide a tool for verification of assumptions
- Assess disaster area and smart city use cases for the application of proposed scalability and physical resource assignment schemes and the verification of testbed and simulation tools

1.3 Literature Review

1.3.1 Sdn scalability

This section summarizes the current work in the literature for SDN scalability. Significant work exists to highlight the optimization of TCAM memory to overcome scalability issues in the literature. Studying existing TCAM optimization strategies is essential for transferring the existing knowledge into the SDN domain. Moreover, the broad application area of SDN is also taken into consideration. To this end, references regarding data center and aerial communication network deployments are also placed within the context of this section.

[10] contains a broad survey regarding perspectives of SDN networks. It analyzes the main challenges faced by SDN. It also provides tables to highlight the contributions of the existing works in literature. [10] categorizes the scalability challenge in SDN in three main aspects. The three major research areas are the performance bottleneck of the SDN Controller, the communication bottleneck between the control and data plane, and limited resources in the data plane. Though these three areas are identified as bottlenecks, only a few studies focus on resolving data plane challenges. It is mainly because the evaluation of modifications on edge devices requires large-scale deployments, and the learning curve of being able to modify switch firmware is high. However, recent publications show a growing interest in addressing the challenges on edge devices by efficient TCAM utilization. [11] also indicates that per-flow control applied in state-of-the-art SDN implementation wastes storage in the data plane and increases delays in flow setup times. A recent survey on SDN challenges is [12]. [12] also indicates the same bottlenecks for scalability in SDN as [10]. Though there is

thorough information regarding the solution on the control plane, there is no significant reference regarding optimization on the data plane.

Recently, the research on TCAM optimization in SDN networks became popular as scalability issue is revealed with actual deployments. TCAM optimization is inspected from several aspects in [13], [14], [15] and [16]. [13] tries to increase TCAM utilization by traffic estimation for the future. This paper also considers the CPU load required for the proposed scheme. Moreover, it provides a heuristic-based computation option to reduce computational costs on the SDN Controller side. However, this method does not consider the cost of TCAM updates. [14] compress the number of rules by utilizing subnet masks. This method also balances the load on the physical links by employing multipath routing methods. However, optimizing subnet masks cannot address the current characteristics of mobile networks. Various flow types are introduced with enormous growth in mobile data rate. [16] suggests that there are 100k flow update requests in data centers. However, optimization of such demand is only feasible on some sides. The computational power required on SDN Controller is huge, and TCAM update cost on OF Switches downgrades the QoE on the user side.

Moreover, the paper suggests default path generation statically and proposes updates upon a topology change. However, this approach is not practical even for content delivery networks. Introducing new containers during rush hours must be handled with a different scheme. The effect of limited TCAM in virtualized data center deployments is highlighted in [17]. [17] states that virtualization in data centers results in overflow in TCAM even before the switch runs out of available physical ports due to the increased number of flows induced by fine-grained control in TCAM.

In [18], the least recently used (LRU) rule is removed from TCAM and stored in SDN Controller. This approach increased the availability of the underlying network. However, the proposed method tends to increase flow setup time for actual deployment. E.g., HTML requests last through a single transaction, while video streams require continuous transmission. When LRU is a video stream, and the new flow is HTML, request redirection of the controller instead of flow installation increases utility with two aspects. First, there is no flow setup delay for HTML. Second, the video stream

will not be removed from OF Table, which reduces flow setup time for the video stream upon the subsequent demand. In addition, this solution is not scalable when the number of switches increases as the storage requirement, and SDN Controller-Switch traffic will grow exponentially.

In [19], aggregation of flows sharing the same destination network is proposed. The aggregation scheme only considers the IP level aggregation for networks. It aggregates the paths by utilizing virtual local area network (VLAN) identifiers. The VLAN requirement reduces the application area of the solution.

[20] suggests that the characteristics of the underlying flows affect the number of rule updates in the network. However, this solution is only tailored for wireless data centers where each OF switch is directly connected to the SDN Controller. In addition, this solution is only evaluated for fat-tree networks, and its applicability for different network topologies remains to be discovered. In addition, this solution increases the CPU load on the OF Switch side by installing multiple rules and employing priorities. This approach tends to downgrade the performance.

In [21], [22] and [23], UAVs are employed in next-generation networks to address several problems such as providing coverage in rural areas without permanent CapEx and providing coverage in disaster areas. In [23], SDN is promoted for orchestrating UAV-based deployment. SDN is picked because of its centralized decision capability. However, [23] requires significant changes to the OF Switch implementation. [23] proposes that OF Switches can store flows to activate them at the pre-determined time frame. The pre-loading scheme targets mitigation of the dynamic structure of UAV networks. However, this certainly reduces scalability and underutilizes TCAMs. [21] targets maximizing the endurance of the UAV network by balancing load over UAVs. [22] tries to optimize the cost of SPs to cover a given area. However, QoE is completely ignored in these papers. Coverage without user satisfaction is acceptable for SPs.

In [24], a direct attack on SDN is explained that exploits the limitation in TCAM resources. The attack generates simple flows to consume TCAM by employing fine-grained control in SDN. [24] emphasizes the content awareness of flow installation strategy on the SDN Controller. Content awareness is required both for security

and efficiency. [25] highlights the attack surface in SDN caused by limited TCAM resources. Furthermore, it also indicates that only limited countermeasures in the existing literature utilize simple traffic patterns. To this end, a more sophisticated flow classification strategy is proposed in [25].

1.3.2 Aerial network deployment strategy

The target application's key performance indicators (KPI) affect the aerial networks' deployment strategy. Energy, coverage, and QoS are the main KPIs used by aerial networks providing temporary network infrastructure like flying base stations or access points.

The primary bottleneck in the aerial network is energy, as stated in [26], [27], [28], [29]. Limited battery power makes it difficult for aerial networks to satisfy lifetime requirements. [26], [27], [29] aims to scale the aerial network application by optimizing power consumption. The [26] aims to increase network lifetime by reducing the number of active UAVs while maximizing coverage. [26] uses a zero overlap strategy for coverage for UAVs to embrace low power consumption. [27] uses battery aware routing scheme to equally distribute the load among network nodes to increase network lifetime. [28] aims to create a path trajectory for UAVs to reach recharge stations, enabling constant recycling of the active UAVs. However, [28] lacks considering QoS requirements of the existing UEs suffering from the disaster. [28] also ignores the active coverage while it focuses on individual UAV's health for battery recovery using recharge stations. [29] aims to deliver connectivity and intelligent appliances to rural areas in developed countries using UAVs. The [29] embraces the microservice architecture and delay-tolerant delivery mechanisms are applied on top of UAV networks to scale the UAV infrastructure for complete internet-of-things (IoT) deployment. [30] splits the deployment area by preventing interference from maximizing coverage with minimal energy consumption by deploying UAVs in three-dimensional (3D) space.

The coverage is highly affected by the density of the deployed UAVs. Dense deployments seriously negatively affect the network lifetime due to excessive power consumption concurrently. In addition, dense deployments also trigger the interference

effect on the underlying wireless communication infrastructure, as given in [31]. [31] promotes using aerial networks for disaster areas due to the decoupling of UAV transportation with ground infrastructure, which is usually seriously damaged during disasters like earthquakes. [26] maximizes coverage area by distributing the UAVs with available batteries to create a square using their circular coverage area. [32] offloads the traffic using aerial base stations (BSs). [32] deploys the UAVs at the same altitude on pre-defined points on the surface and measures the distance with the user equipment (UEs) to define the number of aerial BS for satisfying quality-of-coverage agreement. [33] addresses the optimal coverage problem. In the meantime, [33] also optimizes transmission power on the UAV BS to reduce its detection by adversaries. [34] attacks the coverage problem from the line-of-sight (LOS) aspect. [34] re-evaluates the operational infrastructure requirements upon reception of new transmission from a UE. [34] uses a gradient-descent algorithm to converge the deployment. [34] also assumes that UAVs fly at fixed altitudes and do not consider battery status. In addition to the papers explained [35], [36] also attacks the network coverage problem. [35] uses the random placement of the UAV nodes while [36] uses integer linear programming (ILP) based optimization schemes. There is much effort in the academy to address coverage problems, but the proposed solutions are tailor-made for specific applications. Furthermore, each application requires the same development effort to satisfy environmental and application-specific requirements.

The QoS is the vital KPI to measure the result of the deployment strategy and power-saving precautions from the UE standpoint. To this end, the weight of the UE is higher than the other KPI elements. [37] implements a deep-RL (DRL) based two-layer deployment scheme for the UAV BS. The decision layer of [37] aims to address UAV trajectory to generate optimal distribution for the operational nodes. [37] uses time, position, and active UE requests as state variables while running the inference. [27] uses priority queues to forward the traffic concerning its importance level. [27] achieves over 90% packet delivery rates for high priority traffic. [27] pre-empts the current transition when higher priority traffic is present. In addition, [27] considers low-priority traffic as delay tolerant and does not forward low-priority packets when bandwidth is unavailable. [38] defines the UAV trajectory to meet the desired traffic's

QoS requirement by analyzing the available slices' size. [39] exploits the multipath capability of SDN. [39] mitigates irregularities in LOS in UAV networks due to continuous change in the topology by using multipath load balancing. [40] investigates over five-hundred papers and reveals the diversification of the QoS requirements for the target applications in UAV networks. [40] also highlights the importance of awareness of the different QoS requirements while proposing a holistic architectural deployment solution for the UAV networks.

1.3.3 Aerial communication resource assignment

The third key element of successful UAV deployment is the efficient distribution of the limited network resources for agile UAV networks. The ad-hoc nature of UAV networks forces UAV infrastructure to be secondary to the user of the communication infrastructure. To this end, the resource assignment problem has an additional dimension to consider characteristics of the primary.

[41] targets the disaster areas for UAV deployments. The primary focus of the [41] is to optimize transmission power and the channel selection of the UEs to increase utilization of the UAV BS during disaster times.

[41] utilizes game theory on UAVs to assist UEs in their channel selection process. UEs independently select the transmission power in a distributed manner. However, this scheme only reduces interference amongst the networks. The presentation of increased throughput in [41] is positive, but the overall improvement in the QoS or packet delivery ratio is not available in the numbers. In addition, [41] requires customization of the communication protocol for implementation, which reduces the target audience in disaster areas. [42] aims to increase the number of admitted UEs to increase the quality of experience (QoE) on the user front. The [41] utilizes ILP to switch admitted UEs to increase resource utilization and the number of admitted UEs. [41] depends on the assumption that the demand is available to the UAV endpoint at the time of UE admission. Similarly, [43] assumes that the machine-to-machine (M2M) demands are available before transmission. [43] aims to continue IoT infrastructure without interruption by increasing the number of served nodes in the M2M network in polynomial time.

In [44] UAV network is a slice concerning altitudes. [44] utilizes a probabilistic approach and signal-to-interference-noise-ratio (SINR) to increase the availability of low altitude UAVs to satisfy QoS agreements of the UEs. [45] uses multi-objective optimization for generating potential strategies for disaster areas by considering the affected crowd, the size of the area, available vital resources, and the deployable size of the UAV network. [45] uses a probabilistic approach to generate the top five strategies for implementation. The validity of the selection is highlighted using simulation environments in [45]. The literature used in section 1.3.3 only depends on simulation and lacks the aspect of actual deployments.

Furthermore, the validation of the assumptions for the papers mentioned above, used in section 1.3.3, needs to be included. The testbed validations prove the accuracy of the assumptions. In addition, the literature contains the amendments needed for the initial assumptions. Moreover, integrating the literature becomes a straightforward process for the industry due to existing proof of concept work.

[46] provides a detailed literature summary for the artificial intelligence (AI) solutions on the ANs. [46] highlights that the RL-based solutions in ANs addressing the communication infrastructure focus on the path trajectory and size of the deployed network but needs to pay attention to the resource assignment problem, which is a vital element preventing the cross-scenario application of existing AN solutions.

[47] embraces collaboration between ground control stations (GCS) and UAVs to decide on the trajectory for the UAVs in AN cooperatively. In addition, the UAVs assign the resources by exploiting the global view of the GCS by gaining awareness about the local topology affecting the interference and channel selection for the communication backend in [47].

1.4 Hypothesis

This thesis mainly expands the use case of RL-based solutions in AN solutions. This thesis hypothesizes bringing RL-based solutions to the architectural challenges to scale SDN deployments. In addition, the expansion of RL-based solutions to remove obstacles introduced by the secondary user characteristic of the AN for physical

resource assignment. The post-disaster deployments and smart-city applications of UAVs are further studied to observe the effect of the hypothesis.

1.4.1 Aerial network deployment for disaster areas

The growth in next-generation wireless network deployments accelerated the era of the connected world. However, the mass number of connected mobile devices causes a sudden surge in demand, temporarily paralyzing the network infrastructure. The uninterrupted data flow is vital for increasing the likelihood of accessing the victims.

The communication infrastructure formation in disaster areas requires the quick deployment and wide availability. UAVs reach disaster areas quickly as they are immune to road damage during a disaster. The disaster victims aim to reach out to many people to raise awareness of their vital signs. Ad-hoc WiFi communication facilitates primitives for enabling one-to-many communications by exploiting clear channels. The RL integration generates a decision at the constant time for a UAV deployment problem during disaster time which is NP-hard.

Chapter 4 explains the RL integration for topology control during disasters to managing ad-hoc WiFi AN. The RL model controls the topology using inference, and the model reward frame structure is sensitive to QoS, UE coverage, and power efficiency. Moreover, the PoC scenario is operated on Istanbul Technical University (ITU) campus to train the developed model. The experiment results achieve over 90% packet delivery ratio, over 97% UE coverage, and 0.28KJ/Bit average power consumption.

1.4.2 Opportunistic wifi access for smart-city applications

The commercial availability of UAVs enabled the deployment of ANs. The growth in research and advancements in WiFi technologies ease the integration of AN deployments with COTS devices. The infrastructure deployment of WiFi technology utilizes a crowded industrial-scientific-medical (ISM) band. Moreover, the channel utilization of the underlying infrastructure could be more optimal due vendor-specific nature of the dynamic frequency selection (DFS) implementations in the 2.4GHz band. An efficient WiFi channel access scheme is necessary for increasing the

utilization of the available spectrum by introducing opportunistic nodes to the infrastructure. Therefore, the RL model is integrated into UAV nodes to manage WiFi channel selection addressing physical resource assignment problems. The simulation results given in chapter 5 show 9% improvement for proper channel selection by demonstrating 91% accuracy, compared to the trivial channel scoring-based selection algorithms.



2. AERIAL NETWORK ARCHITECTURE

This chapter defines the architecture of the AN component of the system deployment model given in figure 1.3. The control methodology of the architecture supports both centralized and decentralized deployments. The role of the UAV, GCS, and the data transmitted through the existing links vary depending on the deployment strategy. The rest of this chapter explains the control strategy, decision nodes and decision capabilities, and critical components of centralized and distributed architectures.

2.1 Centralized Aerial Network Deployment

The centralized deployment strategy embraces the SDN approach to orchestrate the network operation. The high-level overview of centralized AN architecture is given in figure 2.1. The SDN controller runs the topology control module and flow-admission policy. The flow-admission policy implements a simple load-balancing algorithm. The pseudo-code of the load-balancing algorithm is given in algorithm 1. The algorithm aims to increase the availability of the UAV flow tables. The algorithm first identifies the UAVs within the proximity of the request. Then the SDN controller equalizes the flow processing delay across the network by balancing the utilization of the flow tables within UAVs. Then the SDN controller redistributes the flows to satisfy the available bandwidth on the wireless interfaces of the UAV nodes. Lastly, the SDN controller synchronizes the flow tables of the active UAVs by using the Micro Air Vehicle Link (MAVLink) protocol on a high-frequency long-range control interface.

The SDN topology control module uses the RL approach and runs the inference on the pre-trained network for maintaining the active network. The SDN controller relies on the quality of the training to maintain an accurate network and provide a long-lasting network experience during AN deployment. The SDN controller aims to provide scalable network deployment service for AN deployments. The scalability depends

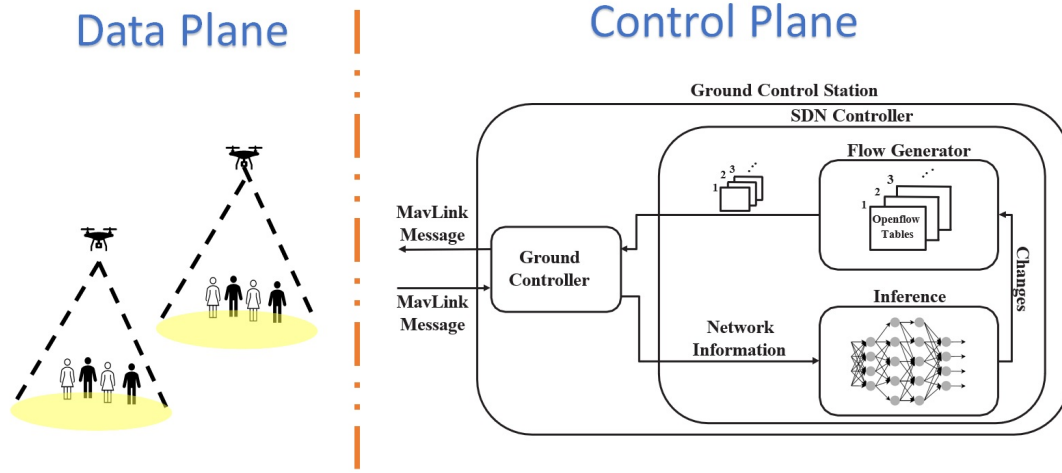


Figure 2.1 : Centralized AN Architecture.

Algorithm 1 Flow admission control algorithm.

Input: Flows, UAVs

```

1: for Flow in Flows do
2:   distClosest = MaxDist
3:   selectedUAV = None
4:   for UAV in UAVs do
5:     if UAV.ftFull then
6:       continue
7:     end if
8:     if !UAV.isActive then
9:       continue
10:    end if distUEtoUAV = dist(flow.UE, UAV)
11:    if distUEtoUAV < distClosest then
12:      if UAV.ftUtil < (selectedUAV.ftUtil × 0.90) then
13:        selectedUAV = UAV
14:      end if
15:    end if
16:  end for
17:  UAV.AddFlow(Flow)
18: end for

```

on the energy consumption, QoS, and coverage of the maintained network during the deployment period.

The UAV node acts as an access entity for the WiFi infrastructure for relaying flows from UEs to the backend. The centralized AN architecture aims to transfer uplink data from UEs to the backend. The sample application requires such transmission to identify the survivors in the disaster areas. The UAV nodes transfer the new flows and

Algorithm 2 Ad-hoc channel selection algorithm.

Input: UAVs, RSSIs

```
1: for UAV in UAVs do
2:   emptyChan = 1
3:   for Chan in Chans do RssiList = RSSIs.getList(UAV)
4:     if RssiList.getRssiFromChan(Chan) < RssiList.getRssiFromChan(emptyChan)
       then
5:       if RssiList.getRssiFromChan(Chan - 1 % 13) < -70 and
         RssiList.getRssiFromChan(Chan + 1 % 13) < -70 then
6:         emptyChan = Chan
7:       end if
8:     end if
9:   end for
10:  if !(emptyChan in OrthogonalChans) then
11:    emptyChan = ClosestOrthogonalChan(emptyChan)
12:  end if
13:  UAV.chan = emptyChan
14: end for
```

flow statistics to the SDN controller using MAVLink. The SDN controller dictates the UAV nodes' position, flow table, and channel assignment.

The UEs utilize WiFi ad-hoc mode to broadcast data in the centralized AN architecture. The UEs scan the environment and select the less occupant channel for broadcasting the data.

The UEs and the SDN controller use the same channel assignment algorithm in algorithm 2.

2.2 Distributed Aerial Network Deployment

Distributed AN architecture aims to remove the command and control interface, which is rarely present on COTS devices. The distributed architecture targets the downlink traffic from the UAV node to GCS. The UAV is the source of the data in this architecture. The GCS collects data from UAVs using an ad-hoc WiFi interface and relays the information to the backbone using wired communication.

Distributed architecture demonstrates the physical resource allocation problem by moving the channel selection to the UAV nodes. Unlike the centralized architecture, the UAV nodes do not have global scan information for the whole environment. The

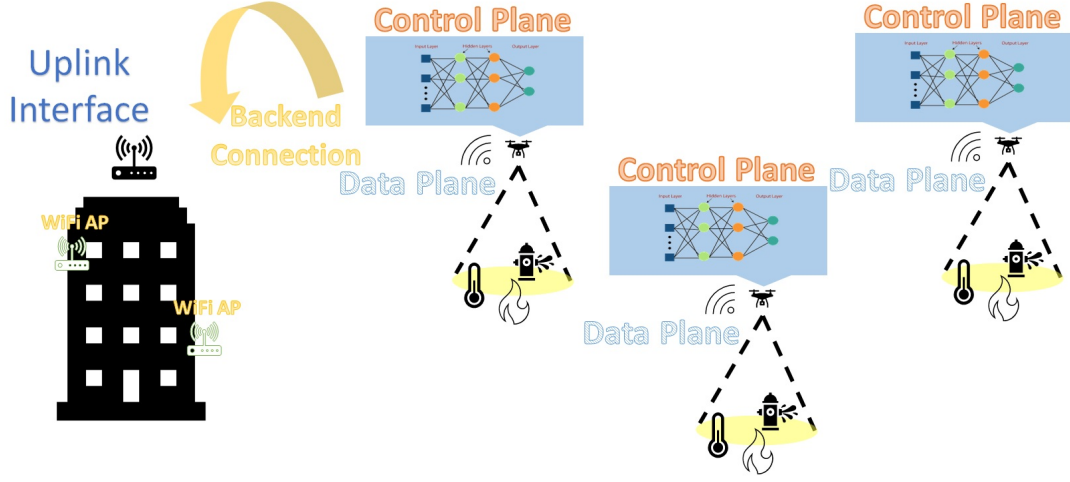


Figure 2.2 : Distributed AN Architecture.

UAVs in distributed architecture are analogous to the UE nodes acting as the data source. To this end, the learnings of the distributed architecture are applicable to enhance the performance of the centralized architecture.

The time, current channel utilization, and 3D location information transform into a probabilistic distribution for the UAV node for the empty channel in the next transmission slot. The UAV node runs inference after sensing the channel allocation. The inference uses weights from the pre-trained dataset. The quality of the dataset increases the decision quality of the UAV for detecting the opportunities in the WiFi environment for sending the available data. The distributed architecture applies to delay-tolerant (DT) UAV sensor networks. Distribution of RL architecture with training weights unifies the estimation efforts in the network. Using a single model removes the inequality in channel utilization throughout the network. Moreover, the RL model is easy to enhance for adding new frequencies as long as data is available to train the network. That is why integrating new wireless technologies becomes a seamless operation within the deployment.

The distributed architecture integrates the RL application into physical resource assignment in AN. Moreover, the new wireless technology integration is enhanced for future provisioning by considering 5GHz expansion and WiFi 6 integration.

2.3 Reward Function

The RL model integrated into AN application requires proper mapping of the reward function for binding the probability distribution of the decision with actual life.

The centralized architecture's topology control RL model rewards QoS, power consumption, and coverage to train the model. The model's training and tuning of the reward function depend on the number of experiments and available data. The centralized model developed in this thesis is pre-trained in a simulation environment for tuning the coefficients of the reward function. The fundamentals of the simulation environment is explained in chapter 3.

The distributed architecture's channel selection algorithm's RL model rewards the selected channel's occupancy and adjacent channels for reducing interference. The channel selection RL model's reward function is tuned using the simulation environment in chapter 3. The public datasets providing WiFi received signal-strength-to-interference (RSSI) given in [48] are used to train and experiment on the network for performance comparison.



3. AERIAL NETWORK SIMULATION ENVIRONMENT

AN architecture simulation is vital due to regulatory limitations on the actual flight capability. The actual flight capability depends on the authorities' approval, weather conditions, the UAV's availability, and the battery's health. However, the adjustment of the reward function for the RL model highly depends on the availability and the variety of the data available. Simulation is the best way to accelerate the development of the model and architecture for ANs. The successful simulation should provide an accurate environment and device-under-test (DUT) model.

3.1 WiFi Environment Simulation

Accurate WiFi data collection is essential for modeling the channel selection algorithm. The OMNET++ discrete event simulator provides a solid infrastructure for collecting data from wireless sensor networks (WSNs) as explained in [49]. The UAVs are analogous to the WSN nodes in smart-city-like applications where telemetry data is transferred to the backend periodically for synchronization.

Furthermore, the targeted 2.4GHz WiFi communication infrastructure is in OMNET++ as per [49]. The gap between real-life deployment and OMNET++-based simulation is the need for dynamic blending capability for the environment on the APs. The dynamic blending of the APs requires dynamic power adjustment and DFS capability. To this end, the DFS mechanism described in [50] is implemented for APs in OMNET++. Moreover, the dynamic power adjustment scheme is enhanced in the OMNET++ to reduce inference across the nodes, improving signal quality and reducing the packet drop due to interference. These two straightforward enhancements narrowed the difference between the WiFi environment in OMNET++ and the real-life deployments.

Furthermore, the AN simulation requires implementing a realistic mobility model for UAVs to complete the data collection chain in the OMNET++. The infrastructure for UAV mobility is described in section 3.2.

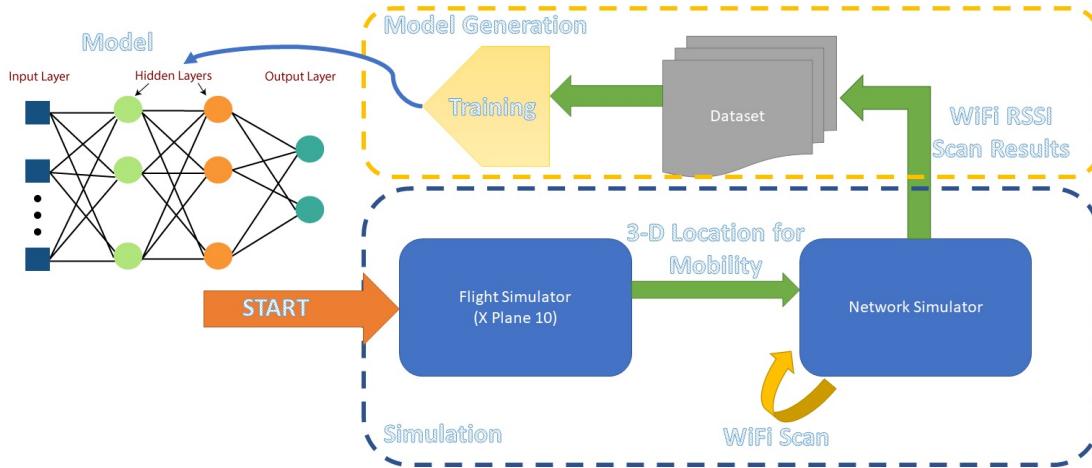


Figure 3.1 : OMNET++ data collection process.

Once the mobility of the UAV's added to the OMNET++, the physical infrastructure is complete for starting data collection using the WiFi scan capability of the WSN nodes already existing in OMNET++'s inet library collection. The data collection routine stores the timestamp, WiFi RSSI data, and 3D position of the node.

The flow and data collection routine within OMNET++ and Aerial Vehicle Network Simulator (AVENS) collaboration is given in figure 3.1.

3.2 UAV Mobility

The mobility model for the UAV and the accurate position information is crucial for adequately mapping the environmental state in the simulation. The flight simulation tools are to educate the aircraft pilots, and their binding to reality is mature. The training and the experience model for the AN require access to the 3D position of the UAV with the timestamp. The AVENS described in [51] developed a plugin for the X-Plane flight simulator to export position information with the timestamp in comma-separated value (CSV) format.

The mobility information generated by the flight simulator is periodically synced with OMNET++ to position the aerial WSN node.

3.3 RL Model Simulation

The behavior of the model needs characterization before application on the UAV nodes. The early characterization is vital due to the cost of hazards and limited flying time availability of the UAVs. To this end, the simulation of the RL model to tune the parameters of the reward function and to observe the behavior of the RL-based decision model is required.

Python provides a rapid development environment for evaluating NN models. The numpy and torch libraries provide the infrastructure for defining NN architecture, activation functions, and error calculation models. In addition to the processing infrastructure, the gym library contains base examples for DRL evaluation which is easy to convert for the target application.

The example contains three major components: NN architecture, environment, and the agent. The environment provides the state of the environment around the agent. It also contains visualization tools for ease of representation. The state variables for the WiFi smart-city application depend on the states' conversion into RSSI sense for all channels around the agent, a UAV node. The agent UAV contains the inference engine, DRL NN architecture, to generate the action as the next channel. The channel update in the agent is the step of the agent. The measurement for the quality of the decision is available upon sensing the environment after the completion of action on the agent side.

Once the environment, agent, and NN setup are complete, the RL simulation environment requires three steps to complete the framework for evaluation:

- The data processing infrastructure: The data conversion module to map inputs to the states. I.e., RSSI sense information around the agent maps into RSSI values for all channels at position X was given time Y.
- Training Loop: The training loop is the fine-tuning between exploration and exploitation within the search domain. Moreover, the reward function is used to tweak the weights of the NN model running the inference given in 4.

- Experiment Loop: This maps into the processing of the available states by the inference engine throughout the available data and the accurate measurement for the decisions generated during the experiment.

Python tool for simulating the disaster area application given in chapter 4 and smart-city application given in chapter ch:SmartNets is available through GitHub via [52].



4. ENERGY-EFFICIENT RL-BASED AERIAL NETWORK DEPLOYMENT TESTBED FOR DISASTER AREAS

4.1 Introduction

The increasing use of advanced 5G networks and mobile services that rely on connectivity has caused a surge in the use of radio frequency (RF) bands. Although this has resulted in a more interconnected world, it has also led to communication issues during disasters. The communication tendency of each device triggers connectivity issues. Regrettably, the RF environment cannot block the unessential traffic from occupying the medium during the disaster. The earthquake in Croatia in March 2020 demonstrates the potential outcome of such an event. The outcome of the surge is given in Figure 4.1 [2] [53]. The AN is viable during disaster times with its independence from roads and capability of generating agile infrastructures temporarily. The temporary infrastructure of AN sustains communication for the affected individuals. The core requirements of disaster areas overlap with the characteristics of AN. The core requirement of AN is an adaptation to constant changes.

This section discusses the current state of research on AN. The research on AN is still evolving and presents many open problems for research. The literature shows that AN has the potential to handle non-persistent traffic but also presents new challenges. Existing research has focused on separating the control layer from UAVs, reducing latency, and simplifying configuration. Moreover, physical network resource allocation still requires further investigation. In addition, the topology control for UAVs is far from real-life deployment due to unrealistic assumptions. This chapter presents a unique approach focusing on physical resource allocation, specifically WiFi channels, and proposes a novel network architecture for disaster areas. This architecture offers zero handover delay, QoS awareness, and energy efficiency. The integration of RL for inference-based decisions allows the deployment of the topology control module on general-purpose computers. The SDN controller orchestrates the

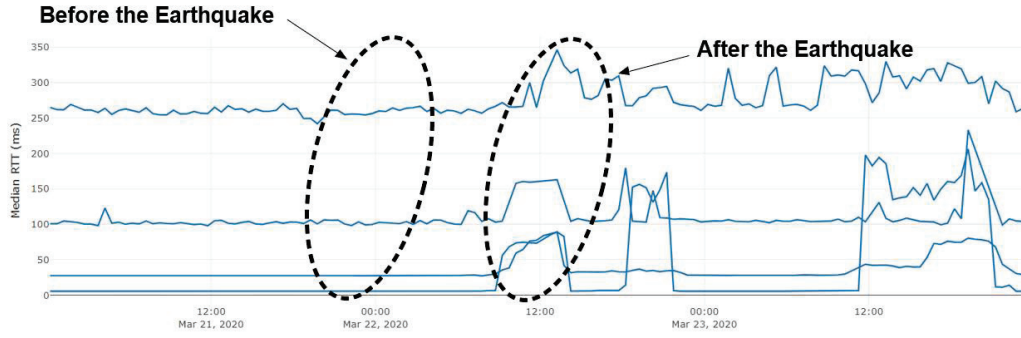


Figure 4.1 : Median RTT value on Croatia Earthquake on March 22, 2020 [2]

synchronization between topology control and flow admission for the underlying network.

Using UAVs for network infrastructures is ideal for handling backend connections in rapidly changing on-demand networks. A growing number of articles focus on AN, such as [54], which proposes an architecture targeting fast convergence while considering the network's delay performance during disaster scenarios. Similarly, [55] highlights the similarities between AN and MANET and VANET and the challenges that AN presents, including 3-D mobility, network convergence time, and limited computing resources for solving complex routing decisions. [56] provides an in-depth performance analysis of AN for complex routing algorithms used in MANETs and VANETs, highlighting the bottleneck on network convergence. To address these issues, [55] suggests utilizing SDN to separate complex routing decisions from UAV nodes in AN and introduce blockchain for the distributed controller layer, thus solving resilience and fail tolerance in the control domain.

Moreover, [57] favors using SDN for AN deployments as it simplifies network configuration for operators by utilizing the Network Function Virtualization (NFV) phenomenon. However, both [55] and [57] fail to inspect physical resource allocation, which is crucial for deploying proposed methods. Additionally, [57] indicates that current hardware and software limitations prevent their proposed method from being realized.

[58] explores using UAV BS and cellular communication nodes to relieve traffic from WAP and decrease queuing delay for Wireless Subscribers (WS). [58] incorporates

using SDN to take advantage of a global perspective and manage spectrum and channel allocation for access nodes. The location of UAV BS, spatial spectrum occupancy, and active flow information within the network are all utilized by [58].

In recent years, there has been increased research on using UAVs for collaborative data collection and processing. Two studies, [59] and [60], have utilized UAVs in data collection. [59] uses AN implementation for resolving the network scarcity problem in IoT networks. UAVs act as relay agents in delay tolerant blockchain to ensure secure data storage with the help of federated learning that depends on iterative data sharing to prevent the leak of private information. Additionally, [60] uses UAVs as surveillance entities with IoT capabilities.

[61] has introduced a dynamic clustering algorithm designed to manage AN deployment. The algorithm utilizes information on the load and battery of the UAV to generate policy output. However, the practical definition of the policy relies on a static model for network parameters such as user mobility and traffic demand rates. This approach limits the algorithm's usability to a specific environment. Moreover, the migration of the solution for new applications becomes impractical.

ANs are suffering from two fundamental problems: UAVs' limited energy and computing capacity. Offloading is a common application utilized by [62] and [63] to reduce power consumption and reducing latency. The [62] and [63] exploit the load information of potential servers. Moreover, [62] and [63] utilize adjacent UAV nodes, cloud servers, and edge servers. [62] and [63] also implement a topology control algorithm to increase the availability of the load balancers within the network. In this context, [64] and [65] achieves the QoS requirements for streaming networks by contributing to the topology management model. [64] and [65] also optimizes power consumption. [66] embraces SDN and optimizes routing tables to reduce latency and improve packet delivery rate within the network.

[67] reduces the overall cost of the network by utilizing UAVs. In addition, [67] also reduces the portion of unutilized resources. The area coverage problem considers the physical aspect of determining the appropriate size for the network deployment.

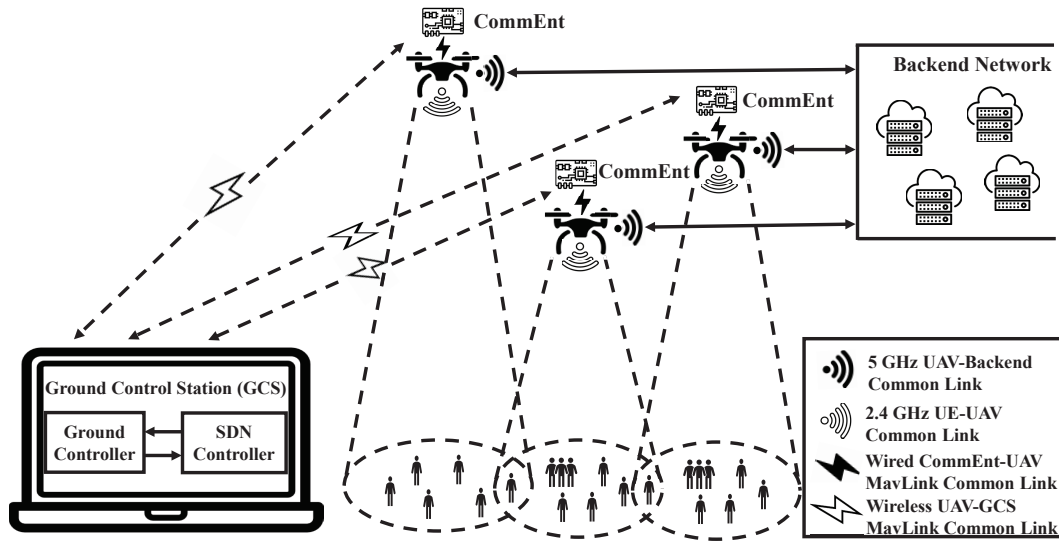


Figure 4.2 : Opportunistic Aerial Network Service Model. [3]

Measured SINR is the QoS indicator for driving the trade-off between available bandwidth for UEs and network size.

The rest of this chapter explains the thesis's contribution to the existing literature. AN disaster area testbed aims to contribute to the KPIs explained in 1.3.2 by implementing the list below:

- Agile opportunistic access network design using WiFi for COTS devices
- The application of RL-based topology control model for UAV networks
- Re-definition of QoS using UE coverage and flow table utilization
- AN testbed deployment

4.2 Architecture

In this section, we present a network architecture that uses UAVs to manage dynamic networks through centralized control, using the SDN approach on the control plane. Figure 4.2 illustrates the overall infrastructure proposed. The architecture intends to address the requirements of the access network in disaster scenarios. Additionally, the backend connection utilizes 5GHz to transfer the traffic using local gateway

entities acting as relay agents for the network. The local backend gateway entities contain redundant communication capabilities such as satellite, cellular, and optical technologies to prevent service disruption due to a single point of failure during the disaster.

The network architecture proposed within this chapter consists of four different entities. The UE is the primary user of the infrastructure. The UE communicates using a 2.4GHz WiFi interface. The traffic source of the architecture is UE for the given architecture. The Communication Entity (CommEnt) bridges UE data with the data backend. CommEnt utilizes WiFi and USB to maintain its communication requirements for the proper BS functionality operation. The CommEnt's WiFi interface uses both 2.4GHz and 5GHz. The ad-hoc mode of WiFi running on both UEs and CommEnt allows disaster victims to communicate through opportunistic channels. The local control entity on UAV is the Flight Management Unit (FMU) which runs an auto-pilot, maintains a 433MHz RF control link, and provides telemetry data for the control plane decisions. 433MHz is used for long-range communication as it does not require any license for communication. The operator of the architecture is the GCS. Section 4.2.1 gives the GCS components' details and functions.

The centralized SDN controller uses telemetry information provided by UAVs of AN. UAV node has an integrated communication entity for providing data to the SDN controller. The GCS controls the topology using the data collected from the UAV nodes. The inference engine of the topology control module runs in the SDN controller of GCS. The central view of the SDN controller allows GCS to distribute channel assignments for the next time slot using the MavLink interface. The channel allocation at each time slot is reported back to the GCS by UAV nodes in the format of received signal strength to interference (RSSI) on the 2.4GHz baseband for every channel stamped with the 3D location.

A MavLink message from UAV highlights the recent transactions for the GCS. This information is vital for the SDN controller to estimate the scale of the current traffic. The GCS visualizes the current topology of the network by collecting battery, position, and flow table load information from UAVs.

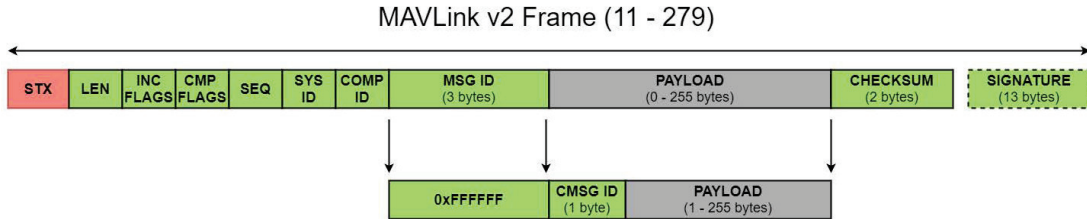


Figure 4.3 : MAVLink Message Format [4]

The GCS executes the decision of the RL model for implementing handovers, load-balancing, and swapping low-battery UAVs as needed. The GCS aims to optimize the number of active UAVs while maintaining desired KPI levels. The optimization of network size is similar to the set-cover problem. The KPI levels the SDN controller of AN targets achieving scalable power consumption performance while maintaining service level agreement (SLA) levels for the UE coverage and QoS.

The physical layout of the network is highlighted in this section. The UAV and the GCS link use the 433MHz MavLink interface via the telemetry module. UAVs provide access to the UEs in the novel agile infrastructure defined in this section. The link between UE and UAV uses a 2.4GHz ad-hoc WiFi link. In addition, UAVs bridge the data plane to the backend through a 5GHz infrastructure WiFi link. FMU placed on the UAV handles auto-pilot tasks and gathers sensor information. The FMU controls the MavLink traffic. The incoming control traffic is relayed to the controller while the telemetry information from the UAV is routed to the GCS. The telemetry information for measuring the health of UAVs contains location and battery.

The reserved MavLink message field enables control communication between GCS and the UAV nodes. The reserved field is used as the "Message Type" of the MavLink protocol, as illustrated in figure 4.3. The details of the custom protocol are given in Table 4.1. The message ID used within standard MavLink is 0xFFFFFFFF.

4.2.1 Ground control station

The ground controller coordinating UAV to GCS communication and the SDN controller orchestrating the network with northbound applications forms the GCS. The ground controller uses TCP to communicate with the SDN controller. The GCS components are illustrated in figure 4.4.

4.2.1.1 Ground controller

Using a USB telemetry module, the ground controller gains the 433 MHz communication interface capability. Moreover, The ground controller acts as an interface between the AN infrastructure and the SDN. The ground controller uses a TCP socket to interface with the SDN controller for providing telemetry data to the SDN controller. The ground controller relays the topology control messages to the UAV nodes using a telemetry module. The ground controller is responsible for delivering the flow-admission control messages.

4.2.1.2 Sdn controller

SDN controller collects the telemetry data from the AN to create the digital twin of the infrastructure. Moreover, the flow table information is also retrieved using custom MavLink messages for displaying and estimating the network load. The network insight is fed into the RL model for triggering topology control decisions for the next time slot.

The topology control decision utilizes 3D location, time, load, UE density, and available battery information. The information of the UAV itself and its closest

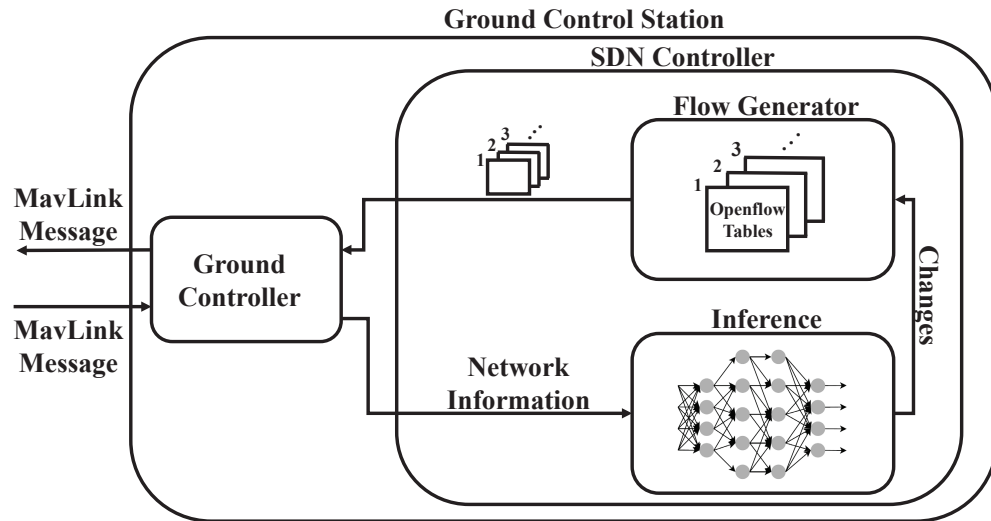


Figure 4.4 : Network Control Modules in Ground Control Station. [3]

neighbor is used as the input of the RL model of the topology control module. Using RL increases the training cost by introducing an additional layer for data collection and model development. However, the inference cost is fixed and constant during the operational state of the network. The fixed computation time provides predictable performance for the SDN topology control module and helps with scaling. The [67] proves the NP-hardness of the set-cover problem which is analogous to the topology control in AN. The use of RL reduces the decision complexity by approximation. The neighbor information is used within the decision to enable a reduction in the network when possible. The load and battery of the neighbor provide an insight into QoS behavior upon charge decision for an active UAV by investigating potential offload opportunities.

Equation 4.1 from [3] proves the analogy of the set-cover problem to the power optimal topology control explained in this section. QoS is opted out of the goal list for ease of explanation. The breakdown of the symbols from the equation 4.1 is as follows:

- $\mathcal{F}$: Set of flow tables
- $\mathcal{F}_i$: Flow table of UAV_i
- A : Operational UAV list
- C : Docked UAV list
- $\mathcal{U}$: Set of UEs
- id_i : The flow sourced from UE_i
- N_{UAV} : Number of UAVs

NP-hard minimum set-cover problem is analogous to the optimal topology control in ANs. The problem 4.1 optimizes a set of active UAVs A by reducing the number of occupied flow tables. Moreover, the problem also includes at least one flow from each UE within active flow tables.

$$\begin{aligned}
& \text{minimize } \sum_{i=1}^{N_{UAV}} \omega_i \mathcal{F}_i \\
& \omega_i : \begin{cases} 1, & UAV_i \in A, \\ 0, & UAV_i \in C \end{cases} \\
& \text{subject to} \\
& id_i \in \mathcal{F}_i \geq 1 \quad \forall \quad id_i \in \mathcal{U} \\
& \mathcal{F}_i \leq 1 \quad \forall \quad i \in \{1, 2, 3, \dots, N_{UAV}\} \\
& \mathcal{F}_i \in \mathcal{F}
\end{aligned} \tag{4.1}$$

The SDN controller runs the inference from an RL model, which is formed as a multi-layer perceptron (MLP). The MLP constructs the NN for the Deep-Q-Network (DQN). As explained in [3], the MLP utilizes telemetry data collected from the UAVs of the AN. The battery, 3D position, and timestamp are the physical components of the input layer (IL). As explained earlier, the physical components are used by the agent UAV and its closest neighbor. The network-level components of the IL are demand and the number of active UEs. The demand is described as the number of flows. The size of the IL is (12+Number of UAVs). The outcome of the RL model is the action for updating the topology in the next step for the target agent. The output layer (OL) is normalized between zero and one. The OL contains the estimation for increasing the reward in the next state. The OL actions are Handover, Callback, and Hold. The layer structure of MLP is size(IL) (input) x (size(IL))² x (size(IL))³ x 3 (OL).

The topology control module evaluates the best action for each UAV by running inference. The estimation of RL for the action is analogous to ω_i from equation 4.1. The RL model is a superset of the optimization problem formalized in equation 4.1. The sensitivity to the battery level and the active UEs increases the complexity of the problem. The weights of the inference engine is defined by training the RL model prior to the deployment. The reward function adjusts the weight of the NN formed as MLP using the reward function in equation 4.3. The reward function is affected by QoS, UE coverage, and power consumption.

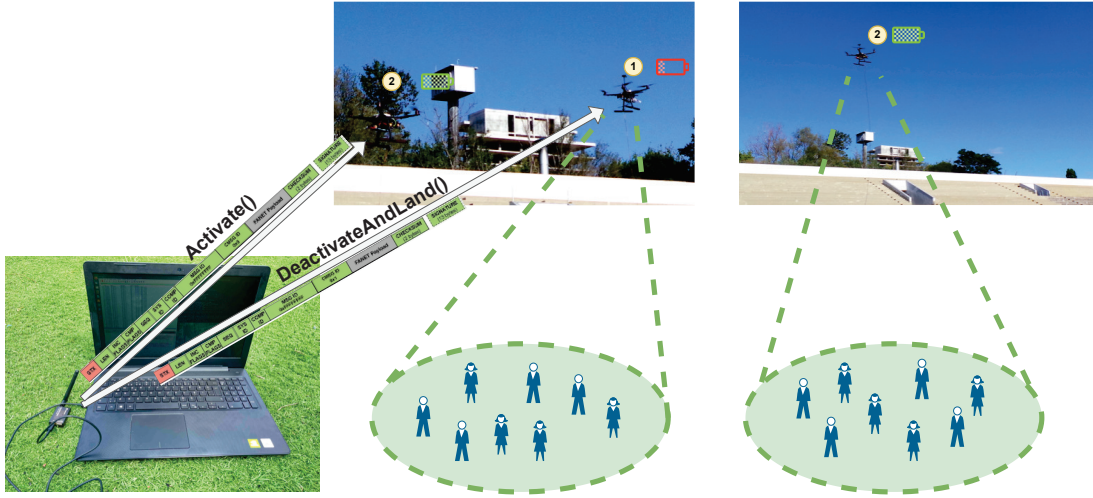


Figure 4.5 : Handover Mechanism. [3]

The flow of the SDN controller to generate topology control actions for active UAVs is given algorithm 3. The training loop consumes the total reward after each episode to adjust weights.

The elements of the reward function in equation 4.3 with their rationale are given below:

- QoS: Packet delivery ratio is used as the QoS of the system. The occupancy of the current flow tables and physical coverage of UE affects the perceived QoS on the UE. The target QoS level is defined as 90% in the SLA for this work. The QoS is normalized to scale to hundred in equation 4.2. The weight of the QoS is used

Table 4.1 : AN Message Types

ID (1 byte)	Content	Direction*	Payload (254 bytes)
0	Flow Install	0	OpenFlow Payload
1	Flow Remove	0	OpenFlow Payload
2	Channel Status	0	Flag** (1 byte) Unused
3	RSSI Values	1	RSSI Info*** x 13 Unused
4	# Active User	1	Unsigned Int (2 bytes) Unused
5	# Active Flow	1	Flow Info**** x 127

* Direction - 0: GCS to UAV, 1: UAV to GCS

** Status - Active(0)/Deactive(1)

*** RSSI Info - (Channel ID(1 byte) RSSI Value(1 byte))

**** Flow Info - (Row ID(1 byte) Flow ID(1 byte))

as 1.3 in reward Equation 4.3. The significance of the QoS is relative to the UE coverage and power consumption in the reward function. The significance of the QoS is empirically defined using simulation. This parameter appears as QoS in the reward equation as well. The mathematical explanation of QoS is given in equation 4.2.

The breakdown of the parameters in equation 4.2 :

- N_{UE} : Number of UEs
- TG_i : Traffic generated by UE_i
- $\mathcal{L}$: Size of the flow table
- $\mathcal{S}_{\mathcal{R}_i}$: Packet statistics of rule i in the flow table
- N_{UAV} : Number of UAVs

$$QoS^* = 100 \times \frac{\sum_{i=1}^{N_{UAV}} (\sum_{j=1}^{\mathcal{L}} \mathcal{S}_{\mathcal{R}_j})}{\sum_{i=1}^{N_{UE}} TG_i} \quad (4.2)$$

- **Power Consumption:** Per-bit power consumption is used for the metric. The measure of power consumption is KJ/bit. The reward for power consumption is derived empirically using simulation o scale the decision quality of the topology control module. The reward breakdown for power consumption is given in equation 4.4.
- **Coverage:** The fairness of the topology is measured by checking the UE with the active flow in active UAVs. This metric is vital for positioning the survivors during the disaster. The coverage contribution has the highest impact on the reward function.

The reward function combining the key metrics explained above is formed to measure the fitness of the topology control decision. The formula for reward is given below:

$$Reward = (QoS \times 1.3) + (Cov \times 1.5) + PowConFactor \quad (4.3)$$

$$PowConFactor = \begin{cases} 110, & \text{if } 0.25 > Pow > 0 \\ 60, & \text{if } 0.30 > Pow \geq 0.25 \\ 40, & \text{if } 0.35 > Pow \geq 0.3 \\ 15, & \text{if } 0.45 > Pow \geq 0.35 \\ 10, & \text{if } 0.60 > Pow \geq 0.45 \\ 1, & \text{if } 1.00 > Pow \geq 0.6 \\ 1/Pow, & \text{otherwise} \end{cases} \quad (4.4)$$

The simulation environment and testbed implementation formed the weights of the critical metrics in the reward function given in equation 4.3.

4.3 Use Cases

The topology control module keeps the current physical infrastructure but updates the physical UAV with the handover decision. The flow of the handover model is explained in section 4.3.1.

Furthermore, the demand increase is satisfied with network expansion when the current deployment falls short in terms of QoS or UE coverage. The process of activating a new cell is explained in section 4.3.2.

4.3.1 Handover mechanism

The handover operation replaces active UAVs to recharge the battery for uninterrupted service on the underlying infrastructure. The network access provider of the UE is altered using handover operation. The flow of the handover execution is given in figure 4.6. Moreover, the capture of handover from testbed implementation is depicted in figure 4.5.

The execution of the handover is triggered with inference output. The substitute UAV downloads the flow table of the operational node. The GCS maintains the download. The clone of the active node travels to the destination. Finally, the GCS finalizes the handover by activating the clone and calling back the deactivated node for charging.

The handover mechanism targets zero disruption in data transfer. To this end, the clone UAV is activated first. Then the deactivation of the operational UAV is completed. The RL model exploits the telemetry data to generate control decisions. The handover

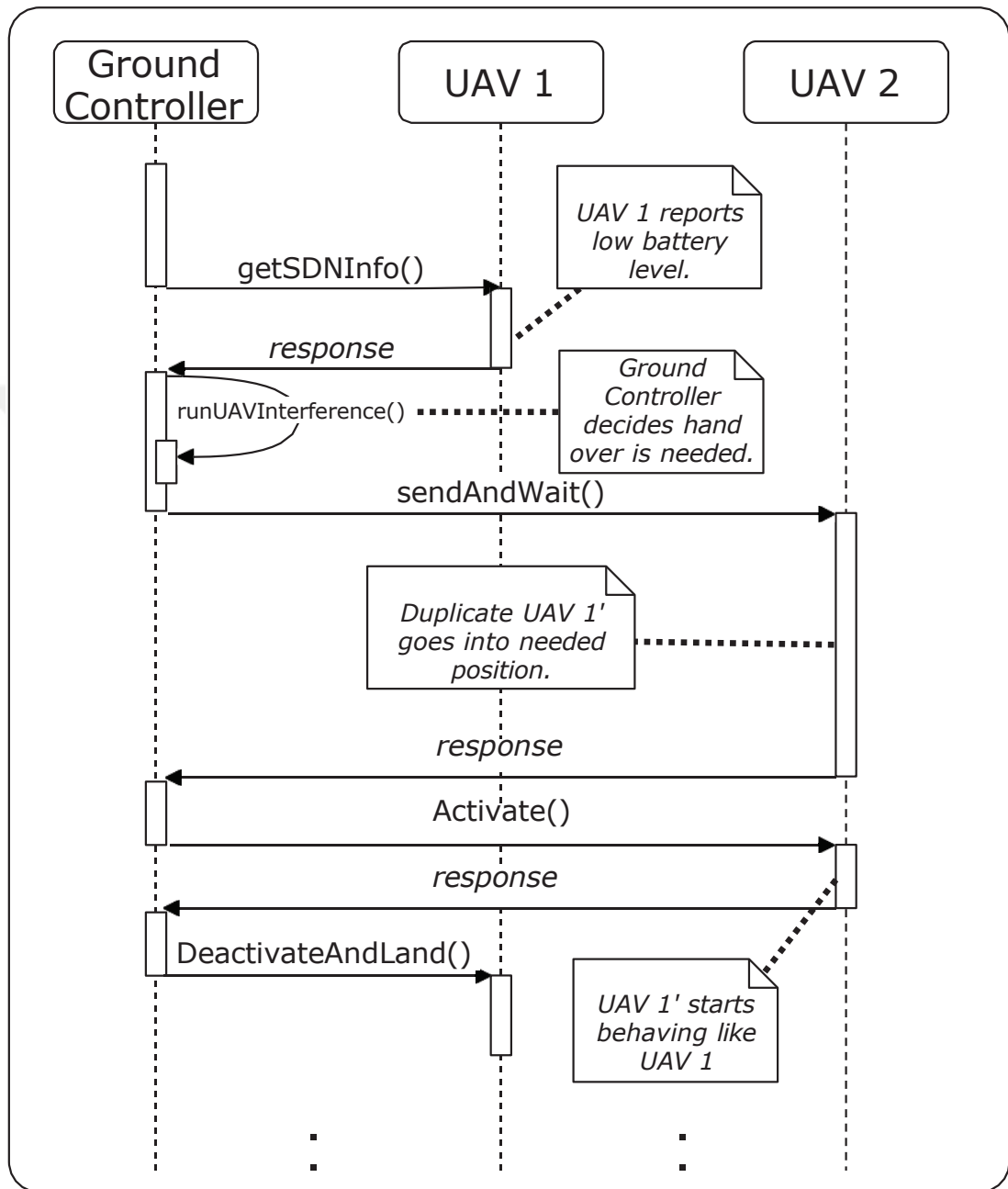


Figure 4.6 : Sample Handover Mechanism Process with Low Battery. [3]

Algorithm 3 Action Generation and Execution

Input: UAVs

```
1: Decisions = new List < Action > ()
2: for UAV in UAVs do
3:   Decisions(UAV) = AN.runUavInference(
4:     Battery,
5:     NumOfActiveFlows,
6:     NumOfActiveUsers,
7:     Location.X,
8:     Location.Y,
9:     Location.Z,
10:    Neighbor.Battery,
11:    Neighbor.NumOfActiveFlows,
12:    Neighbor.NumOfActiveUsers,
13:    Neighbor.Location.X,
14:    Neighbor.Location.Y,
15:    Neighbor.Location.Z,
16:    Foreach UAV in UAVs UAV[UavIdx].IsActive)
17: end for
18: for Decision in Decisions do
19:   if Handover then
20:     Dup = SDN.Duplicate(UAV)
21:     SDN.SendAndWait(Dup)
22:     SDN.Activate(Dup)
23:     SDN.DeactivateAndLand(UAV)
24:   else if Callback then
25:     SDN.DistributeFlows(UAV)
26:     SDN.DeactivateAndLand(UAV)
27:   else if Hold then
28:     Donothing
29:   end if
30: end for
31: WiFiChannel = AN.SelectChannel(UAVs.RssiList)
32: SDN.DistributeChannel(Channel)
```

decision triggers the selection of a replacement UAV. In addition, the flow table is duplicated from the operational UAV. Then the UAV node access to the closed perimeter. The GCS enables the activation of clone UAV and deactivates the active UAVs in order. Finally, the deactivated UAV is called back to recharge the battery.

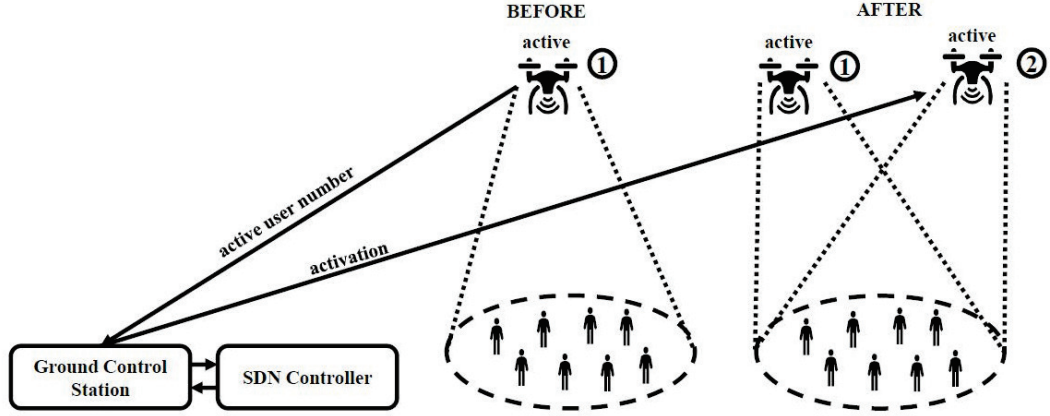


Figure 4.7 : New Cell Deployment Scenario in Theory [3]

4.3.2 New cell deployment

The SDN controller uses the RL engine to deploy a new UAV to the field. The SDN controller optimizes QoS over power consumption by adjusting the number of UAVs in AN. The telemetry data stored in the SDN explained in 4.2.1.2 empowers the basis for the decision. The algorithm 3 exploits the retrieved data to generate topology control decisions for the next time slot. The demonstration of new cell deployment is given in figure 4.7 and figure 4.8. Adjusting the network size mandates the callback of UAVs if the demand decreases in the access network.

4.4 Performance Evaluation

This section evaluates the proposed model's efficacy and scrutinizes the outcomes to affirm the solution's validity.

4.4.1 Testbed setup

The testbed deployment contains 4 UAVs. The UAVs are equipped with Lithium Polymer (LiPo) batteries and manufactured in China. The 25 minutes of flight time depends on the weather conditions. The control entity of the UAV is Raspberry Pi 4 Computer Model B [68]. The customized UAV is depicted in figure 4.9. The Open vSwitch converts Raspberry Pi into an OF switch. The OF switch uses [69]. The

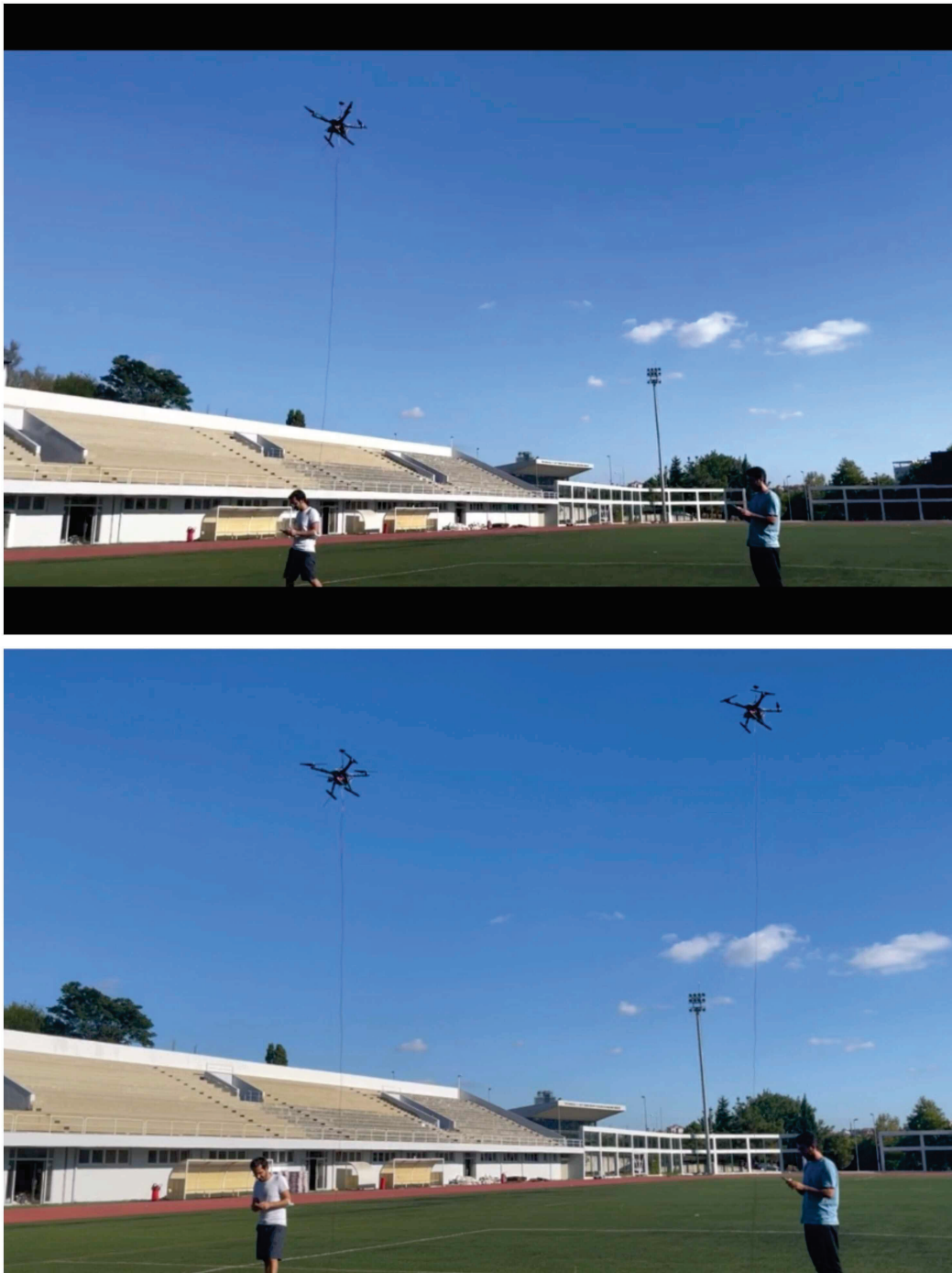


Figure 4.8 : New Cell Deployment Scenario in Action [3]

v2.15.0 version of OpenVSwitch is employed by definition. The host running the SDN controller is still under discussion. The GCS runs on Ubuntu 21.04 operating system.

Raspberry Pi selection relies on the specification given below:

- Supports WiFi,
- 4GB memory,
- Quad-Core 1.5GHz ARM-v8 processor.

The power of serviced solutions enables efficient execution of inference with MLP. The device also runs on limited power without the performance hit. The main components of the testbed in terms of tooling and equipment are as follows:

- Raspberry Pi - FMU Link: Raspberry Pi is equipped with MavLink protocol [70] using MAVSDK Library given in [71]
- UAV - GCS Link: MavLink protocol
- FMU Autopilot: FMUK66-based PixHawk autopilot firmware [72]
- Traffic Source: Packet Sender [73] software is used for packet generation
- Flow Frequency: 3 to 25 flows from each Raspberry Pi
- Flow Class: TCP, UDP, and VoIP and randomly distributed

The GCS specification and setup is given below:

- Intel(R) Core(TM) i7-1065G7 CPU
- 16 GB RAM
- Windows 10 Pro operating system
- Two different VM container
 - AN to GCS interaction using MavLink through USB telemetry module
 - SDN controller runs topology control and flow-admission protocol using ONOS given in [74]

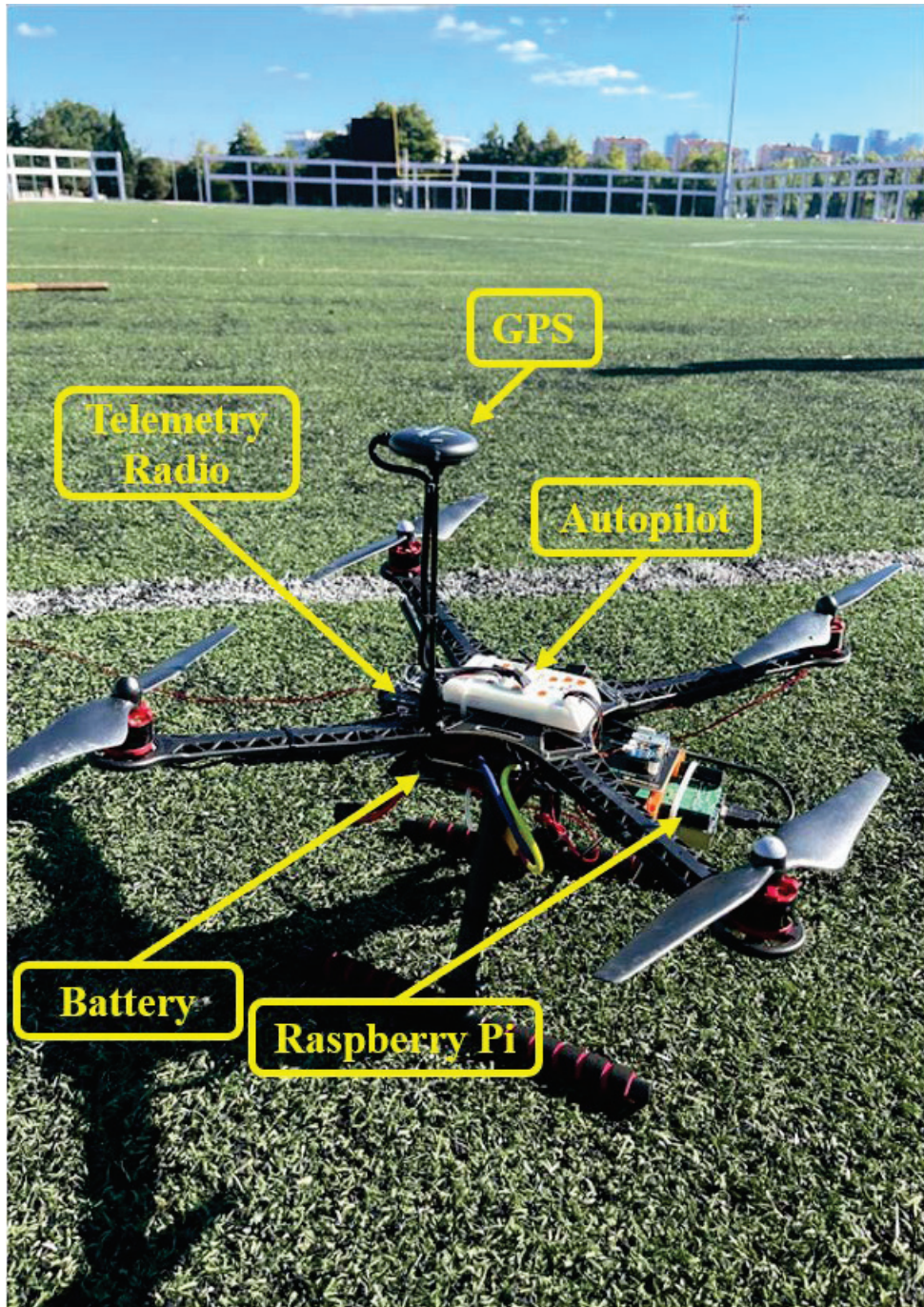


Figure 4.9 : Test-bed setup on the Drone. [3]

- Ubuntu 20.04.2 LTS operating system

The testbed data is used to train the RL model developed for topology control. The RL model is placed within the SDN controller as highlighted in 4.2.1.2. The episode of the training is divided into periods of 128 discrete timeslots. The topology control module uses algorithm 3 to generate action for each agent. The RL model is validated

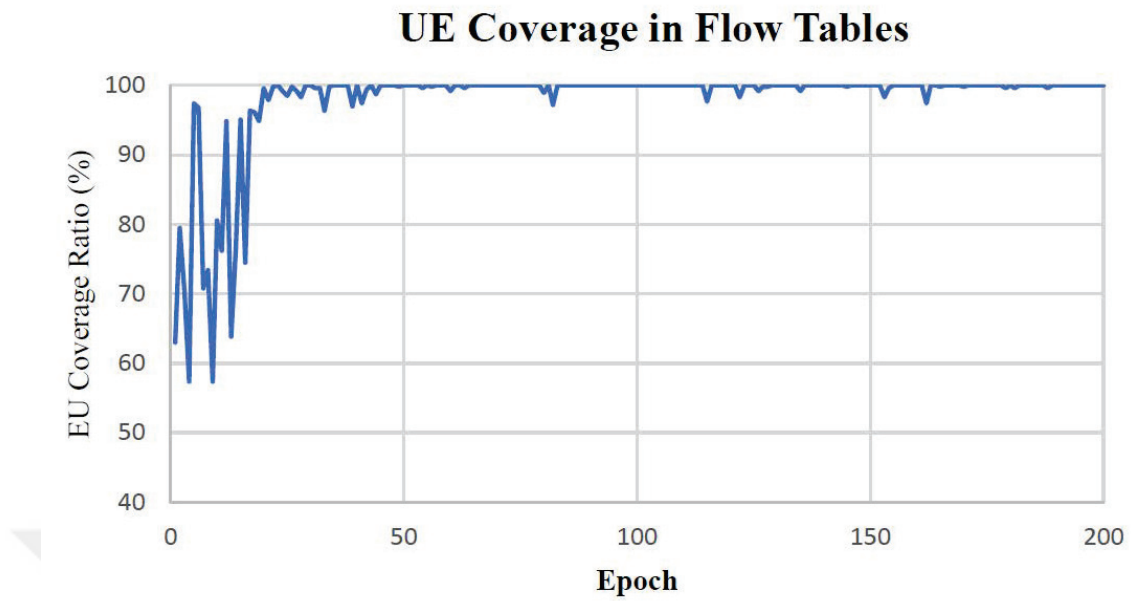


Figure 4.10 : UE Coverage trend in testbed-based training of ML model. [3]

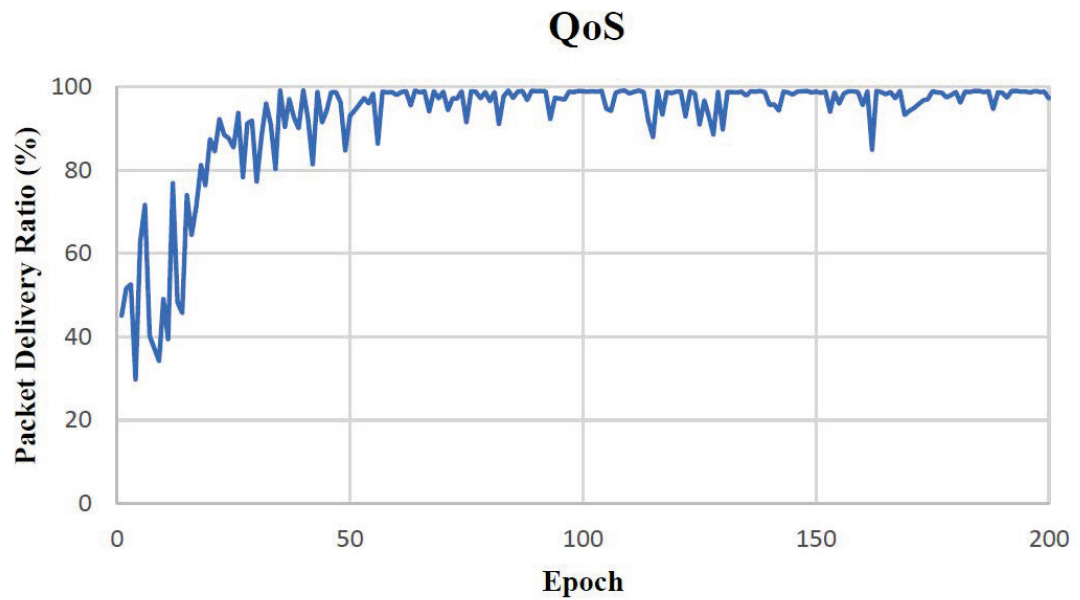


Figure 4.11 : QoS, packet delivery ratio, the trend in testbed-based training of ML model. [3]

by observing the training trend, QoS, UE coverage, and power consumption data from ONOS.

4.4.2 Performance results

The significance of parameters used in the reward function defines the order of convergence during training. The results of the experiment depicted in this section validate the control strategy of the RL model formed using DQN.

The UE coverage reaches maximum coverage in 40 episodes as highlighted in figure 4.10. The Figure 4.10 represents the $id_i \in \mathcal{F}_i$ from equation 4.1.

Furthermore, the model converges QoS to exceed the SLA value of UEs. As depicted in figure 4.11, the QoS exceeds 90% during training on the testbed and meets the KPI criteria. The approximation on QoS takes 80 episodes in the training loop. Both figure 4.10 and figure 4.11 demonstrate oscillation in the performance, which is generated due to random walks in the search space. The random walks prevent approximation to the local maxima and expand the RL model's discovery capability. The frequency of oscillation reduces with time as the RL model learns and increases the exploitation rate in its decision process. The last achievement of the model is stable per-bit power consumption as illustrated in figure 4.13.

Moreover, the RL model proposed in this chapter scales in terms of power consumption, mitigating the increased number of UEs. The RL model also mitigates the increased packet delivery ratio by keeping power consumption at 0.28KJ/Bit throughout the experiment. However, the [54] is tailor-made for specific applications, and the generated traffic significantly affects power consumption.

Figure 4.12 reveals the iterative improvement in the decision quality. As explained earlier, the decision quality is represented with a reward function favoring QoS, UE coverage, and power consumption. The RL model approximates 94% of the potential reward.

The AN control strategy defined in this chapter outperforms [67] in terms of power consumption and network lifetime. The comparison in terms of scaling reflected as power consumption is given in table 4.2.

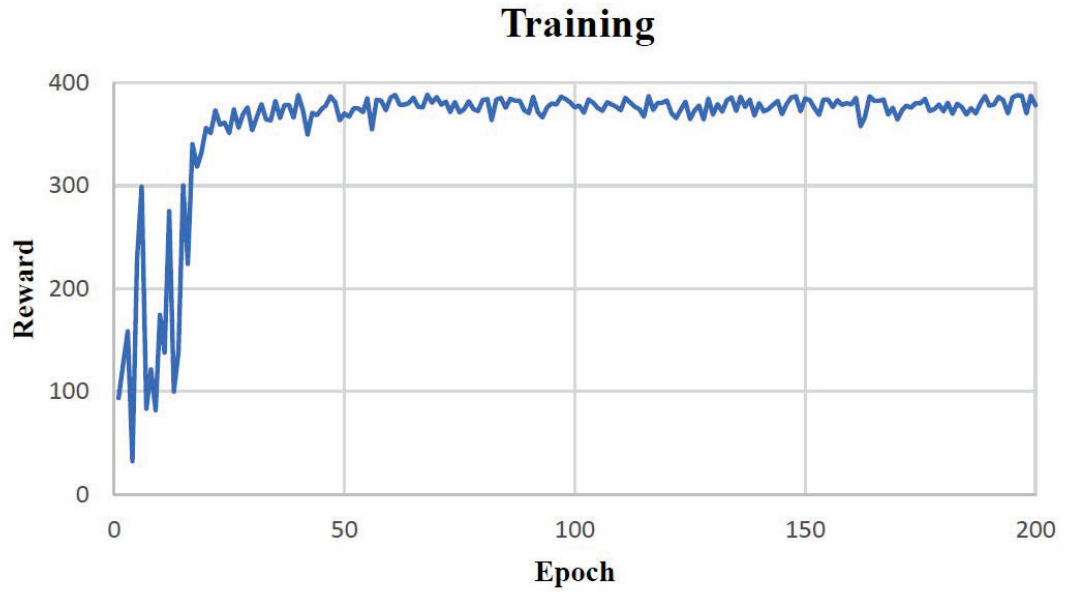


Figure 4.12 : Reward trend of testbed-based training of ML model. [3]

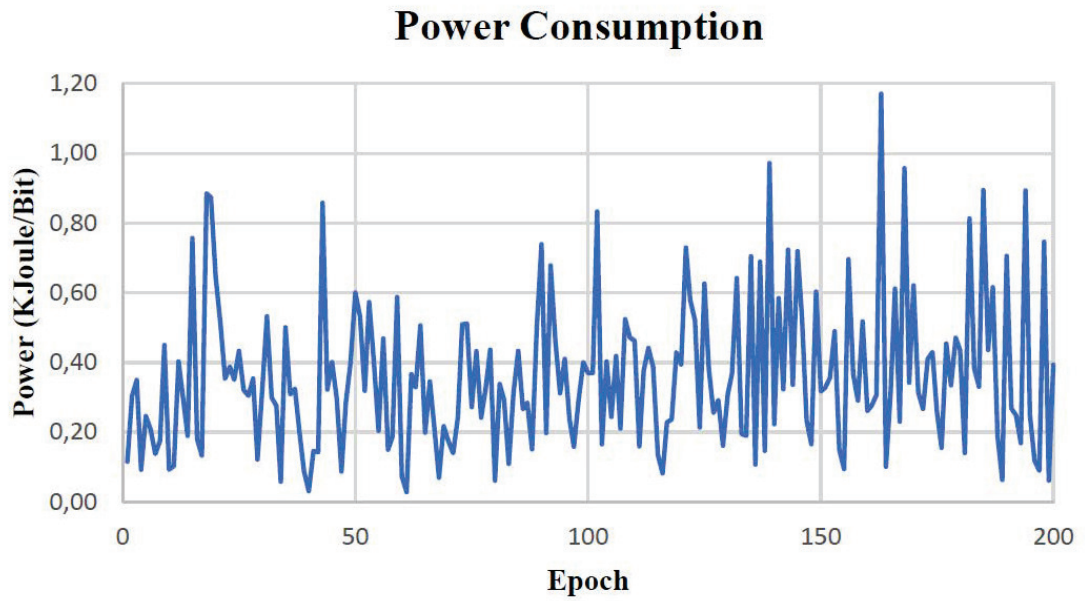


Figure 4.13 : Power consumption trend in testbed-based training of ML model. [3]

The perception of coverage is quite different between our method and [67]. The [67] aims to increase area coverage and deploys unnecessary UAVs to the field even when the UE is not on the site.

Table 4.2 : Performance Comparison

Method	Coverage	QoS	Power
Proposed	97%	$\geq 90\%$	0.28KJ/bit
Method [67]	98%	$\geq 70\%$	0.42KJ/bit

The definition of QoS is also different between our method and [67]. The [67] aims to increase SINR. However, the UE density is ignored, which reduces QoS with saturation on the physical resources on the access network.

5. OPPORTUNISTIC RL-BASED WIFI ACCESS FOR AERIAL SENSOR NODES IN SMART CITY APPLICATIONS

5.1 Introduction

Smart city applications are becoming widespread. Mobile environmental sensing is one of the critical applications for smart city deployments. The mobile sensing solution deployment is accelerated with reduced equipment cost and increased availability of COTS UAV nodes. The use of UAVs in such applications is well researched in literature as given in [75]–[78].

The WiFi provides potential ease of deployment for mobile UAV sensors with its broad availability. However, the existing crowd on the WiFi infrastructure doubts the performance and viability of the AN sensor deployments.

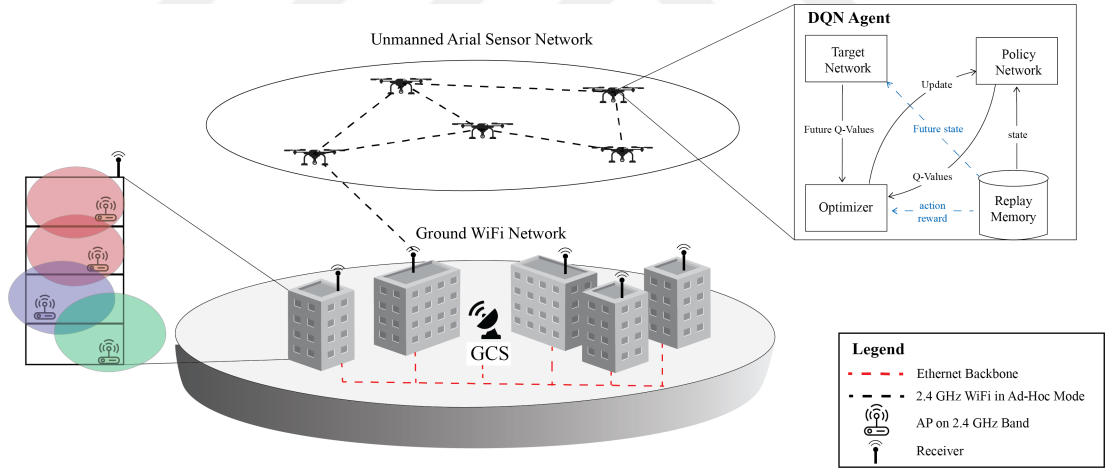


Figure 5.1 : System Model. [5]

The crowded WiFi environment requires DFS implementation to reduce the interference in the environment to increase the ratio of successful transmissions. This problem is similar to the k-coloring problem for the graphs. The k-coloring problem is NP-hard by definition. To this end, resolving the optimal channel in constant time takes work. The similarity of the channel selection problem is well explained in [79].

The AP vendors embrace the utilization of orthogonal channels in their vendor-specific implementations. The channel scoring mechanism marks the best channel available, and the closest orthogonal channel is picked for the next time slot, as explained in [50]. The vendor-specific nature of this solution reduces the scalability and effectiveness of the solution in the field. In addition, using non-orthogonal channels enables opportunistic users to send data to the WiFi medium.

The integration of ML has been highly considered in literature in recent years. The duty cycle optimization, beamforming management, and interference mitigation are addressed in [80]–[82].

Though there is an effort on physical resource management with RL, most of the current work focuses on the UAV ANs' path trajectory. The [83] develops and trains a DRL model for mitigating latency and reducing interference by adjusting transmit power on the planned path. [84] optimizes the transmission power of the UAVs and couples the adjustment decision with the coordinate on the path for uplink and downlink communication. The current transmit power optimization work also explains the opportunistic nature of the UAV networks by aiming to stay away from interference. Power optimization reduces the operation area of the UAV by limiting the communication range.

This chapter integrates the RL model into the physical resource allocation for a 2.4GHz WiFi network. The main contribution of this chapter to the thesis is given below:

- AN for mobile sensors is defined
- Ad-hoc WiFi communication is utilized for opportunistic communication
- RL model is integrated for WiFi channel selection
- Simulation environment is developed for opportunistic WiFi channel selection for AN with mobile sensors

5.2 System Model

The Unmanned Aerial Sensor Network (UASN) and the Ground WiFi Network are defined in the top-level architecture and illustrated in figure 5.1.

5.2.1 Unmanned aerial sensor network

The UASN incorporates UAVs and a GCS communicating through 2.4 GHz WiFi in Ad-Hoc Mode.

The specification of UAVs is listed below:

- WiFi Radio: 802.11ac and Ad-hoc mode support
- Single Board Computer: Inference operation and WiFi control
- Load: Under 500 gr to comply with regulations the Remote Control: Mission configuration using Sub-1GHz

The computational load on the distributed architecture is reduced by utilizing offline training. Offline training also removes the necessity for single-board computers with large memory. The RL model exploits the available channels for the AN mobile sensor node to deliver the telemetry data through GCS.

The GCS is equipped with multiple WiFi cards to handle concurrent data retrieval from mobile sensors. The backend interface of the GCS is maintained using the ethernet connection. The top-level architectural model of the system for concurrent data acquisition system in smart cities is depicted in figure 5.1.

5.2.2 Ground wifi network

The crowd in the WiFi network is generated by the APs deployed in the buildings and on the streets to enable residents' access through their mobile devices and personal computers. The interference generated by the infrastructure APs is circumvented using an RL-based channel selection model within this chapter. The opportunistic access

or aerial nodes increases the utilization of the available spectrum for the underlying network.

5.3 Opportunistic Reinforcement Learning Model

The UAV agents become immune to the changes in the WiFi environment by using the DRL model for selecting the channel for data transfer. The quality of the decision and the immunity to the change depending on the available data and the time spent on the training for the model.

The DRL model uses 3D position of the UAV and occupancy of the WiFi channel in the form of RSSI. The agent's behavior improves by exploitation of the rewards.

5.3.1 Environment

The input data for the RL model is prepared by sensing the environment using a WiFi radio. The WiFi radio scans the environment and stores the RSSI values for each channel by stamping the data using 3D position and the acquisition time. The data format is given in table 5.1.

Table 5.1 : Environment Data

Label	Description
3-D Location	Latitude, Longitude, Altitude
Timestamp	Date and Time
Selected Channel	Channel #
Channel Occupancy	RSSI [-120 dB, 0]dB

5.3.2 Actions

The UAV agent runs inference at each transmission period for delivering data to the backend using a WiFi channel. The RL model generates a probability of non-occupancy for each 20MHz channel at the 2.4GHz WiFi band. The agent switches to the channel with the highest probability of allowing successful transmission of the sensor data.

5.3.3 Deep q network

The DQN incorporates two separate networks during training. The target network is the currently active network used in inference for exploitation, while the policy network is instantly updated with every operation. The weights of the policy network are transferred to the target network periodically to measure the reward of the active weight throughout an experiment.

The representation of the policy network is $PN(s_t, a_t, \theta)$. The notation expands into action a_t applied at time t in the environmental state s_t . The NN is formed as MLP with two hidden layers. The formation of NN is IL (16) $\times$ hidden layer I (256) $\times$ hidden layer II (256) $\times$ OL (14). The input layer utilizes the WiFi environment around the UAV agent. The activation of hidden layers is triggered with ReLu, and the output layer generates the probabilities of each action in section 5.3.2.

The notation of the target network is similar to the policy network, as expected. The target network is represented as $TN(s_t, a_t, \theta')$ where the θ and θ' are the weights stored in matrice format.

The Adam optimizer used in the training of the RL model uses the loss function given below:

$$L(\theta) = E[R(s_t, a_t) + \gamma \max_{a \in A} TN(s_{t-1}, a_t, \theta') - PN(s_t, a_t, \theta)^2] \quad (5.1)$$

where the $R(\cdot)$ represents the collected reward with RL response in state s_t with action a_t . The empirically derived reward function given in equation 5.2 is used for generating the reward output.

$$Reward = Q + N + B \quad (5.2)$$

where

$$Q = \begin{cases} 0, & \text{if } 0dB \leq RSSI(C) < -30dB \\ 5, & \text{if } -30dB \leq RSSI(C) < -67dB \\ 20, & \text{if } -67dB \leq RSSI(C) < -80dB \\ 24, & \text{if } -80dB \leq RSSI(C) < -90dB \\ 30, & \text{if } -90dB \leq RSSI(C) < -120dB \end{cases} \quad (5.3)$$

where C is the selected channel

$$N = \begin{cases} 0, & \text{if } 0dB \leq RSSI(N) < -67dB \\ Q/2, & \text{if } -67dB \leq RSSI(N) < -80dB \\ Q-2, & \text{if } -80dB \leq RSSI(N) < -120dB \end{cases} \quad (5.4)$$

where N is the adjacent channel

$$B = \begin{cases} 15, & \text{if } RSSI(C) = MIN(RSSI) \\ 0, & \text{otherwise} \end{cases} \quad (5.5)$$

The replay memory enables batch training for the RL model to mitigate instability given in section 5.4. The representation of replay memory is $(s_t, a_t, R(s_t, a_t), s_{t+1})$.

5.4 Model Training

5.4.1 Data requirements

The RL model developed for channel selection requires 3D coordinates as the location and channel utilization of the environment around the UAV. The PoC data collection environment is developed using the infrastructure implemented in chapter 4. The PoC data collection environment is explained in section 5.5.1. The setup is used to verify the assumptions to differentiate from the existing literature. The existing literature contains unrealistic assumptions that prevent the proposed solutions' implementation in reality. The RL model is evaluated using the data retrieved from a public data set available as given in [76]. The synchronization of GPS and UAV time is essential for accurately training the model from the available dataset.

5.4.2 Training method

The training loop given in algorithm 4 is operated using the data retrieved from the public dataset. The environment for running the simulation requires Python 3.9 and pyTorch 2.0.0 with 500 MB of memory.

The policy network is fed with each potential action in the given state. The state is the 16-tuple containing each channel's 3D position, time, and RSSI information as explained in section 5.3. The rewards are calculated using the target network, and the weight of the target network is synchronized with the policy network periodically.

The agent prevents over-fitting by utilizing the replay memory and running batches represented as C using a policy network to diversify the search space. This mechanism is also illustrated in figure 5.1 with blue markers. The policy network is only used during training, and the deployed network only utilizes the target network to run inference.

Algorithm 4 DQN Training Loop

```

1: ReplayMemoryCapacity =  $C$ 
2: ExperienceMemory = new Experience( $C$ )
3: PolicyNetwork = DQN()
4: TargetNetwork = DQN()
5: Clone PolicyNetwork weights into TargetNetwork
6: for  $e$  in episodes do
7:   State =  $s$ 
8:   for  $t$  in time do
9:     Action = Agent.SelectAction()
10:    Reward = Environment.TakeAction()
11:    NextState = Observation.GetNext()
12:    NewExp = Experience(State, Action, Reward)
13:    ExperienceMemory.Push(NewExp)
14:    if ExperienceMemory.EnoughSamples() then
15:      Batch = ExperienceMemory.Sample()
16:       $Q$  = PolicyNetwork.Calculate.Q(Batch)
17:       $Q'$  = TargetNetwork.Calculate.Q'()
18:      Loss = MSELoss( $Q$ ,  $Q'$ )
19:      Optimizer = Adam.ZeroGradient(Loss)
20:      PolicyNetwork.Step(Optimizer)
21:    end if
22:    if  $t \bmod \text{UpdateInterval}$  then
23:      TargetNetwork = PolicyNetwork
24:    end if
25:  end for
26: end for

```

5.5 Benchmark

The benchmark of the RL model is obtained using the public dataset with ID4 from [48]. The RL channel selection is compared with vendor-specific DFS implementation, random channel selection, and fixed channel assignment. The details of the vendor-specific channel selection for Arista APs is given in [50].

5.5.1 Test setup

The PoC testbed is implemented in ITU using the infrastructure developed in [3]. The validity of the model and assumptions are proved with testbed implementation. However, training the model requires a significant amount of data which means additional time and permission. To this end, the benchmark is held using [48], which provides a comprehensive dataset with 1775 APs and a single UAV equivalent.

Evaluation is performed using a Python-based simulation framework ported from the topology control model used in chapter 4. The performance assessment of four different channel estimations is compared with the tool. The underlying method for each strategy is explained in section 5.5. The flow of the simulation algorithm is given in algorithm 18. The 1% fixed packet loss is applied, and the data packet transmission is emulated for each decision for four different strategies.

The KPI of the RL model is the accuracy of the channel estimation. A two-fold approach is used for rewarding. The best channel selection is the first pillar, while the proper channel selection is also appreciated within the evaluation. The definition of the proper channel is the channel having noise level, which would not hijack the data transmission.

The definition of accuracy is given in equation 5.6. The $Num(S)$ in the equation represents successful identification of the best channel in the 2.4GHz WiFi spectrum. The $Num(E)$ represents the number of steps exists in the dataset.

$$Accuracy = (Num(S)/Num(E)) \times 100 \quad (5.6)$$

Algorithm 5 Evaluation Loop

```
1: for state in Dataset do
2:   time = state.GetTime()
3:   position = state.GetPosition()
4:   channelRssiList = state.GetRssiList()
5:   actionRL = ASN.runInference.(
6:     time,
7:     position.X,
8:     position.Y,
9:     position.Z,
10:    Foreach channel in channelRssiList channel.rssi)
11:   actionApVendor = ASN.runChannelScore
12:   actionRandom = RandomInteger(1,13)
13:   actionFixed = fixedChannel
14:   for Method in Methods do
15:     SetChannel(action[Method])
16:     SendData()
17:   end for
18: end for
```

5.5.2 Performance analysis

The effect of hyper-parameter tuning to understand the importance of the training and the performance of the integrated RL model is evaluated in this section. The test results are given in section 5.5.1. The RL model performance is compared against vendor-specifics DFS, random channel selection, and fixed channel selection strategies.

The learning rate used for the DQN affects the uphill reward trend by shifting the exploitation in the search space. When the learning rate is increased, and the uphill trend of the reward is accelerated, the RL model tends to miss the local optima and loss diverse application capability.

The learning rate is updated during simulation to measure the effect of the hyper-parameter. Figure 5.2 illustrates the disparity between the learning rate of 1/1000 and 1/100 using reward response by inspecting the trends in the moving average of 10 episodes. The 1/1000 learning rate explores the search space more extensively, as observed from the saw in the trend-line in the graph. The 1/1000 learning rate also generates a better reward response in the long run. In figure 5.3, the impact of the exploration is evident as the learning rate of 1/1000 achieves a 46% success rate

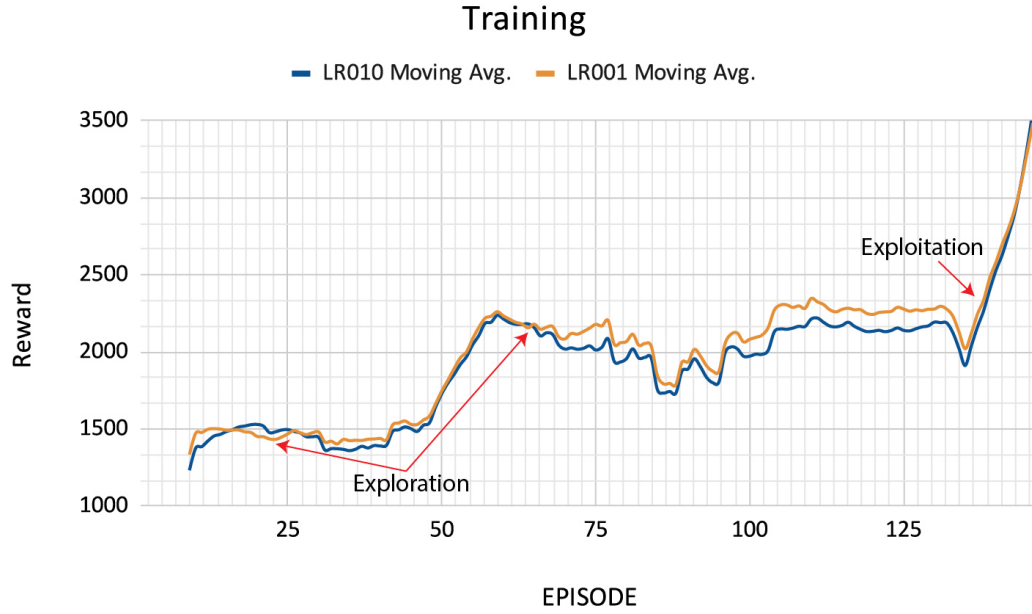


Figure 5.2 : Training results for the given network. [5]

compared to the 17% success rate of the 1/100 learning rate when searching for the best available channel. Though the overall performance of the 1/100 learning rate is worse than the learning rate 1/1000, it still provides a similar experience with vendor-specific DFS solutions.

The RL model outperforms the competition by providing 91% transmission opportunity for the sensor nodes, which is 9% better than the current vendor-specific DFS deployments. The comparison between the four strategies implemented is depicted in figure 5.3. The vendor-specific DFS achieves 80% accuracy while random selection provides 55% opportunity for the sensor nodes.

The quality of the RL model is differentiated significantly when the best-channel selection is assessed within the results. The RL performs 5.75 fold better performance compared to the second-best option. The RL-based models are immune to environmental changes and are easier to develop than the stationary solution as long as data is available.

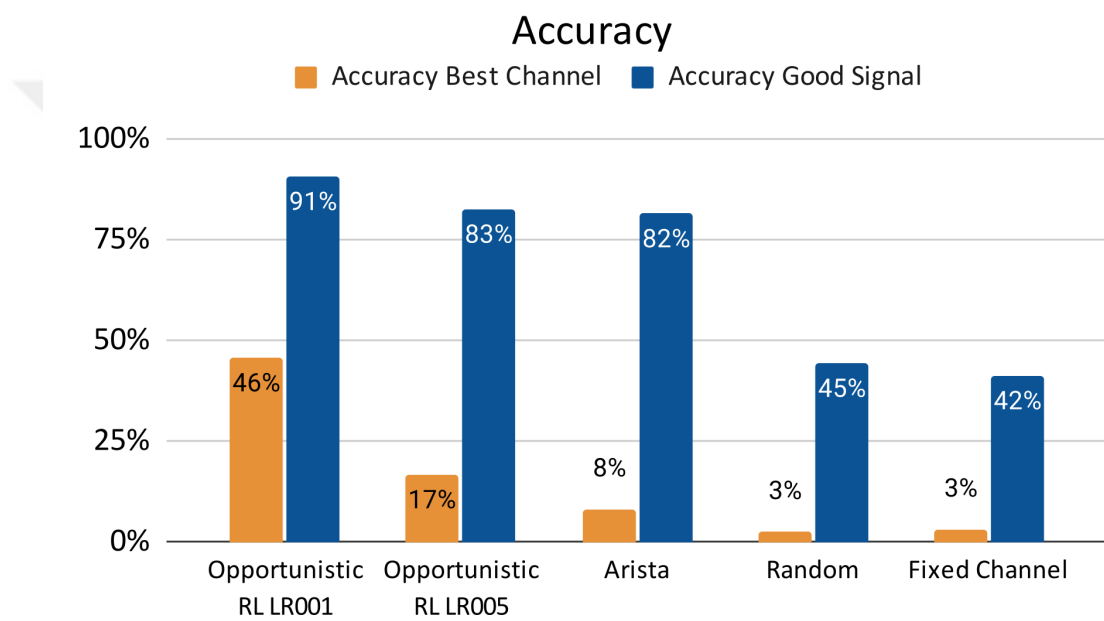


Figure 5.3 : Performance comparison of the proposed method. [5]



6. CONCLUSIONS

This chapter provides brief information about findings from applications. This chapter also lists the tools developed for evaluating the AN in simulation format. The thesis is finalized by providing insight into future work to enhance the findings of this work in section 6.3.

The simulation tool for integrating RL into AN is developed using Python and available in [52]. The topology control module using the UAV environment and physical resource selection module estimating WiFi channel are integrated into the simulation.

The findings of centralized architecture for integrating the RL model into AN application for disaster area is described in section 6.1. In addition, the findings of the distributed RL integration in AN is explained in section 6.2

6.1 Findings of Centralized Aerial Network Deployment

The centralized AN explained in 4 provides a new system for disaster zones that uses SDN technology to manage the AN. This system aims to improve user coverage, maintain high QoS standards, and optimize energy efficiency (measured in KJ/Bit). Unlike the approach described in [54], our method offers a more scalable power consumption scheme by providing stable per-bit consumption rates regardless of the amount of data processed.

6.2 Findings of Distributed Aerial Network Deployment

The distributed RL integration for AN is proposed for the Opportunistic RL-based WiFi access scheme for aerial sensor nodes to use WiFi resources efficiently. For this, the DQN agents are designed for physical resource identification on radio resources deployed in UAVs. As a result, the proposed scheme outperformed traditional methods by 9% in excellent signal quality selection and 5.75 fold in best channel selection. For

future work, motivated by the lack of available data in this area of research, we plan to implement comprehensive campus data collection at ITU.

6.3 Future Work

The comprehensive data collection efforts in ITU campus for 5GHz and 2.4GHz WiFi infrastructure are needed to accelerate RL integration training efforts to address physical resource assignment challenges in AN.

Integrating distributed architecture for channel selection into disaster areas for improving applicability is needed to remove MavLink control from GCS to increase the operation range of disaster area deployment.

The Research on RL integration for flow-admission control is needed for enhancing independence in disaster area deployment.

REFERENCES

- [1] FAA Aerospace Forecast Fiscal Years (2022), **Technical Report**, Federal Aviation Administration, https://www.faa.gov/sites/faa.gov/files/2022-06/Unmanned_Aircraft_Systems.pdf, accessed on: 2023-03-07.
- [2] *Internet Health Report*, <https://ihr.iijlab.net>, accessed: 2021-08-31.
- [3] Ariman, M., Akkoç, M., Sari, T.T., Erol, M.R., Seçinti, G. and Canberk, B. (2023). Energy-efficient RL-based aerial network deployment testbed for disaster areas, *Journal of Communications and Networks*, 25(1), 25–34.
- [4] Project, D. (2021). *MAVLink Packet Serialization*, <https://mavlink.io/en/guide/serialization.html>, accessed on: 2021-08-19.
- [5] Ariman, M., Cakir, L.V., Ozdem, M. and Canberk, B. (2023). Opportunistic RL-Based WiFi Access for Aerial Sensor Nodes in Smart City Applications, *2023 International Conference on Smart Applications, Communications and Networking (SmartNets) (SmartNets 2023)*, Istanbul, Turkey, p. 5.
- [6] *Commercial Drone Industry Trends (2018)*, <https://www.dronedeploy.com/resources/ebooks/2018-commercial-drone-industry-trends-report/>, accessed on: 2023-05-06.
- [7] OpenFlow Switch Specification v1.3 (2012), **Technical Report**, OpenNetworkFoundation, <https://www.opennetworking.org/wp-content/uploads/2014/10/openflow-spec-v1.3.0.pdf>, accessed on: 2023-04-05.
- [8] Cisco Visual Networking Index: Forecasts and Trends, 2017-2022 (2019), **Technical Report**, Cisco, <https://twiki.cern.ch/twiki/pub/HEPIX/TechwatchNetwork/HtwNetworkDocuments/white-paper-c11-741490.pdf>, accessed on: 2023-05-04.
- [9] *Hyperdense HetNets: Definition, drivers and barriers (2012)*, https://scf.io/en/documents/180_Hyperdense_HetNets_Definition_drivers_and_barriers.php, accessed on: 2023-05-06.

- [10] **Benzekki, K., El Fergougui, A. and Elbelrhiti Elalaoui, A.** (2016). Software-defined networking (SDN): a survey, *Security and Communication Networks*, 9(18), 5803–5833.
- [11] **Huang, N., Li, Q., Lin, D., Lit, X., Shen, G. and Jiang, Y.** (2018). Software-Defined Label Switching: Scalable Per-Flow Control in SDN, pp.1–10.
- [12] **Saraswat, S., Agarwal, V., Gupta, H., Mishra, R., Gupta, A. and Dutta, T.** (2019). Challenges and Solutions in Software Defined Networking: A Survey, *Journal of Network and Computer Applications*.
- [13] **Tian, Y., Chen, W. and Lea, C.** (2018). An SDN-Based Traffic Matrix Estimation Framework, *IEEE Transactions on Network and Service Management*, 15(4), 1435–1445.
- [14] **Zhang, S.Q., Zhang, Q., Tizghadam, A., Park, B., Bannazadeh, H., Boutaba, R. and Leon-Garcia, A.** (2017). TCAM space-efficient routing in a software defined network, *Computer Networks*, 125, 26 – 40, <http://www.sciencedirect.com/science/article/pii/S1389128617302645>, softwarization and Caching in NGN.
- [15] **Afek, Y., Bremler-Barr, A., Feibish, S.L. and Schiff, L.** (2018). Detecting heavy flows in the SDN match and action model, *Computer Networks*, 136, 1 – 12, <http://www.sciencedirect.com/science/article/pii/S1389128618300859>.
- [16] **Zhao, G., Xu, H., Chen, S., Huang, L. and Wang, P.** (2017). Deploying default paths by joint optimization of flow table and group table in SDNs, *2017 IEEE 25th International Conference on Network Protocols (ICNP)*, pp.1–10.
- [17] **Johnson, S.** (2013). *Exploring OpenFlow Scalability in Cloud Provider Data Centers*, <https://searchnetworking.techtarget.com/feature/Exploring-OpenFlow-scalability-in-data-centers>.
- [18] **Marsico, A., Doriguzzi-Corin, R. and Siracusa, D.** (2017). Overcoming the memory limits of network devices in SDN-enabled data centers, *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pp.897–898.
- [19] **Zhang, X., Yu, S., Zhang, J. and Xu, Z.** (2019). Forwarding Rule Multiplexing for Scalable SDN-Based Internet of Things, *IEEE Internet of Things Journal*, 6(2), 3373–3385.
- [20] **Chuang, C., Yu, Y. and Pang, A.** (2018). Flow-Aware Routing and Forwarding for SDN Scalability in Wireless Data Centers, *IEEE Transactions on Network and Service Management*, 15(4), 1676–1691.

- [21] **Zhang, X., Wang, H. and Zhao, H.** (2018). An SDN framework for UAV backbone network towards knowledge centric networking, *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp.456–461.
- [22] **Chiaraviglio, L., Amorosi, L., Blefari-Melazzi, N., Dell’Olmo, P., Natalino, C. and Monti, P.** (2018). Optimal Design of 5G Networks in Rural Zones with UAVs, Optical Rings, Solar Panels and Batteries, *2018 20th International Conference on Transparent Optical Networks (ICTON)*, pp.1–4.
- [23] **Barritt, B., Kichkaylo, T., Mandke, K., Zalcman, A. and Lin, V.** (2017). Operating a UAV mesh and internet backhaul network using temporospatial SDN, *2017 IEEE Aerospace Conference*, pp.1–7.
- [24] **Cao, J., Xu, M., Li, Q., Sun, K., Yang, Y. and Zheng, J.** (2018). Disrupting SDN via the Data Plane: A Low-Rate Flow Table Overflow Attack, pp.356–376.
- [25] **Xu, T., Gao, D., Dong, P., Foh, C.H. and Zhang, H.** (2017). Mitigating the Table-Overflow Attack in Software-Defined Networking, *IEEE Transactions on Network and Service Management*, 14(4), 1086–1097.
- [26] **Gao, F., Zhou, Y., Ma, X., Yang, T., Cheng, N. and Lu, N.** (2019). Coverage-maximization and Energy-efficient Drone Small Cell Deployment in Aerial-Ground Collaborative Vehicular Networks, *2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, pp.559–564.
- [27] **Ma, T., Li, B. and Nie, G.** (2021). An Energy Saving Routing Strategy with Differentiated Service Provide Ability for UAV Assisted Disaster Rescue, *2021 7th International Conference on Computer and Communications (ICCC)*, pp.64–69.
- [28] **Golam, M., Akter, R., Lee, J.M. and Kim, D.S.** (2021). Energy Efficient UAV Deployment with Optimized Path-Planning in Post-Disaster Environment, *2021 International Conference on Information and Communication Technology Convergence (ICTC)*, pp.466–469.
- [29] **Galán-Jiménez, J., Vegas, A.G. and Berrocal, J.** (2022). Energy-efficient deployment of IoT applications in remote rural areas using UAV networks, *2022 14th IFIP Wireless and Mobile Networking Conference (WMNC)*, pp.70–74.
- [30] **Babu, N., Papadias, C.B. and Popovski, P.** (2020). Energy-Efficient 3-D Deployment of Aerial Access Points in a UAV Communication System, *IEEE Communications Letters*, 24(12), 2883–2887.
- [31] **Sánchez-García, J., García-Campos, J., Toral, S.L., Reina, D.G. and Barrero, F.** (2015). A Self Organising Aerial Ad Hoc Network Mobility Model for Disaster Scenarios, *2015 International Conference on Developments of E-Systems Engineering (DeSE)*, pp.35–40.

- [32] **Savkin, A.V. and Huang, H.** (2019). Deployment of Unmanned Aerial Vehicle Base Stations for Optimal Quality of Coverage, *IEEE Wireless Communications Letters*, 8(1), 321–324.
- [33] **Fan, J., Zhang, L., Shen, G., Li, Z., Cao, P. and Lei, L.** (2020). A Self-deployment Algorithm for Flying Ad Hoc Networks (FANETs) towards Low Probability of Detection Coverage in Directional Antennas, *2020 IEEE 20th International Conference on Communication Technology (ICCT)*, pp.898–902.
- [34] **Savkin, A.V., Huang, C. and Ni, W.** (2023). On-Demand Deployment of Aerial Base Stations for Coverage Enhancement in Reconfigurable Intelligent Surface-Assisted Cellular Networks on Uneven Terrains, *IEEE Communications Letters*, 27(2), 666–670.
- [35] **Sharma, V., Patel, R.B., Bhaduria, H.S. and Prasad, D.** (2015). Policy for random aerial deployment in large scale Wireless Sensor Networks, *International Conference on Computing, Communication Automation*, pp.367–373.
- [36] **Hassan, S.S., Park, Y.M. and Hong, C.S.** (2021). On-Demand MEC Empowered UAV Deployment for 6G Time-Sensitive Maritime Internet of Things, *2021 22nd Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pp.386–389.
- [37] **Zhou, Y., Ma, X., Hu, S., Zhou, D., Cheng, N. and Lu, N.** (2022). QoE-Driven Adaptive Deployment Strategy of Multi-UAV Networks Based on Hybrid Deep Reinforcement Learning, *IEEE Internet of Things Journal*, 9(8), 5868–5881.
- [38] **Cho, J.W., Yang, P., Quek, T.Q. and Kim, J.H.** (2020). Service-Aware Resource Allocation Design of UAV RAN Slicing, *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, pp.801–805.
- [39] **Singhal, C. and Rahul, K.** (2019). Efficient QoS Provisioning Using SDN for End-to-end Data Delivery in UAV Assisted Network, *2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, pp.1–6.
- [40] **Baltaci, A., Dinc, E., Ozger, M., Alabbasi, A., Cavdar, C. and Schupke, D.** (2021). A Survey of Wireless Networks for Future Aerial Communications (FACOM), *IEEE Communications Surveys Tutorials*, 23(4), 2833–2884.
- [41] **Su, Z., Dai, M., Xu, Q. and Li, R.** (2020). A Novel Resource Allocation Scheme With Unmanned Aerial Vehicles in Disaster Relief Networks, *2020 IEEE/CIC International Conference on Communications in China (ICCC)*, pp.102–106.

- [42] **Nguyen, M.T. and Le, L.B.** (2021). Resource Allocation, Trajectory Optimization, and Admission Control in UAV-Based Wireless Networks, *IEEE Networking Letters*, 3(3), 129–132.
- [43] **Liu, X. and Ansari, N.** (2019). Resource Allocation in UAV-Assisted M2M Communications for Disaster Rescue, *IEEE Wireless Communications Letters*, 8(2), 580–583.
- [44] **Helmy, M. and Arslan, H.** (2018). Utilization of aerial heterogeneous cellular networks: Signal-to-interference ratio analysis, *Journal of Communications and Networks*, 20(5), 484–495.
- [45] **Diao, L., Liu, Y., Liu, W. and Chiaraviglio, L.** (2022). Unmanned Aerial Vehicle Assisted Healthcare Resource Allocation in Disasters, *2022 32nd International Telecommunication Networks and Applications Conference (ITNAC)*, pp.1–5.
- [46] **Lakew, D.S., Tran, A.T., Masood, A., Dao, N.N. and Cho, S.** (2023). A Review on AI-Driven Aerial Access Networks: Challenges and Open Research Issues, *2023 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC)*, pp.718–723.
- [47] **Zhao, J., Sun, X., Ma, X., Zhang, H., Yu, F.R. and Hu, Y.** (2023). Online Distributed Optimization for Energy-Efficient Computation Offloading in Air-Ground Integrated Networks, *IEEE Transactions on Vehicular Technology*, 72(4), 5110–5124.
- [48] **Cruz, S. and Aguiar, A.** (2020). *Cooperative Localization in Vehicular Networks Dataset*, <https://dx.doi.org/10.21227/jtqe-q772>, accessed on: 2023-04-22.
- [49] **Varga, A. and Hornig, R.** (2008). An Overview of the OMNeT++ Simulation Environment, *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems Workshops*, Simutools '08, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), Brussels, BEL.
- [50] **Networks, A.** *Automatic and Dynamic Channel Selection*, <https://wifihelp.arista.com/post/automatic-and-dynamic-channel-selection/?pdf=229>, accessed on: 2023-04-22.
- [51] **Marconato, E., Rodrigues, M., Pires, R., Pigatto, D., Querino Filho, L., Pinto, A. and Castelo Branco, K.** (2017). AVENS – A Novel Flying Ad Hoc Network Simulator with Automatic Code Generation for Unmanned Aircraft System.
- [52] **Ariman, M. and Erol, M.R.** (2022). *WiFi UAV RL Evalution Tool*, https://github.com/mehmetariman/UAV_RL_Research.git, accessed on: 2023-05-24.

- [53] *RIPE Labs, After the Quake in Croatia*, <https://labs.ripe.net/author/becha/after-the-quake-in-croatia/>, accessed: 2021-08-31.
- [54] **Kaleem, Z., Yousaf, M., Qamar, A., Ahmad, A., Duong, T.Q., Choi, W. and Jamalipour, A.** (2019). UAV-Empowered Disaster-Resilient Edge Architecture for Delay-Sensitive Communication, *IEEE Network*, 33(6), 124–132.
- [55] **Hu, N., Tian, Z., Sun, Y., Yin, L., Zhao, B., Du, X. and Guizani, N.** (2021). Building Agile and Resilient UAV Networks Based on SDN and Blockchain, *IEEE Network*, 35(1), 57–63.
- [56] **Rosati, S., Kruzelecki, K., Heitz, G., Floreano, D. and Rimoldi, B.** (2016). Dynamic Routing for Flying Ad Hoc Networks, *IEEE Transactions on Vehicular Technology*, 65(3), 1690–1700.
- [57] **Nogales, B., Sanchez-Aguero, V., Vidal, I. and Valera, F.** (2018). Adaptable and Automated Small UAV Deployments via Virtualization, *Sensors*, 18(12).
- [58] **Ali, M.A., Zeng, Y. and Jamalipour, A.** (2020). Software-Defined Coexisting UAV and WiFi: Delay-Oriented Traffic Offloading and UAV Placement, *IEEE Journal on Selected Areas in Communications*, 38(6), 988–998.
- [59] **Islam, A., Al Amin, A. and Shin, S.Y.** (2022). FBI: A Federated Learning-Based Blockchain-Embedded Data Accumulation Scheme Using Drones for Internet of Things, *IEEE Wireless Communications Letters*, 11(5), 972–976.
- [60] **Islam, A., Rahim, T., Masuduzzaman, M. and Shin, S.Y.** (2021). A Blockchain-Based Artificial Intelligence-Empowered Contagious Pandemic Situation Supervision Scheme Using Internet of Drone Things, *IEEE Wireless Communications*, 28(4), 166–173.
- [61] **Asaamoning, G., Mendes, P. and Magaia, N.** (2021). A Dynamic Clustering Mechanism With Load-Balancing for Flying Ad Hoc Networks, *IEEE Access*, 9, 158574–158586.
- [62] **Shukla, R.M., Sengupta, S. and Patra, A.N.** (2018). Software-defined network based resource allocation in distributed servers for unmanned aerial vehicles, *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, pp.796–802.
- [63] **Ashraf Ateya, A.A., Muthanna, A., Kirichek, R., Hammoudeh, M. and Koucheryavy, A.** (2019). Energy- and Latency-Aware Hybrid Offloading Algorithm for UAVs, *IEEE Access*, 7, 37587–37600.
- [64] **Cumino, P., Lobato Junior, W., Tavares, T., Santos, H., Rosário, D., Cerqueira, E., Villas, L.A. and Gerla, M.** (2018). Cooperative UAV Scheme for Enhancing Video Transmission and Global Network Energy Efficiency, *Sensors*, 18(12).

- [65] **Zhao, Z., Cumino, P., Souza, A., Rosário, D., Braun, T., Cerqueira, E. and Gerla, M.** (2019). Software-defined unmanned aerial vehicles networking for video dissemination services, *Ad Hoc Networks*, 83, 68–77, <https://www.sciencedirect.com/science/article/pii/S1570870518306231>.
- [66] **Dapper e Silva, T., Emygdio de Melo, C.F., Cumino, P., Rosário, D., Cerqueira, E. and Pignaton de Freitas, E.** (2019). STFANET: SDN-Based Topology Management for Flying Ad Hoc Network, *IEEE Access*, 7, 173499–173514.
- [67] **Huang, H., Huang, C. and Ma, D.** (2020). A method for deploying the minimal number of UAV base stations in cellular networks, *IEEE/CAA Journal of Automatica Sinica*, 7(2), 559–567.
- [68] *Raspberry Pi 4* (2021), <https://www.raspberrypi.org/>, accessed on: 2021-08-31.
- [69] *Open vSwitch* (2021), <https://www.openvswitch.org/>, accessed on: 2021-08-31.
- [70] *MavLink Protocol* (2021), <https://mavlink.io/en/>, accessed on: 2021-08-31.
- [71] *MAVSDK Library* (2021), <https://mavsdk.mavlink.io/main/en/index.html>, accessed on: 2021-08-31.
- [72] *FMUK66 based PixHawk Autopilot Firmware* (2021), <https://github.com/PX4/PX4-Autopilot>, accessed on: 2021-08-31.
- [73] *Packet Sender* (2021), <https://packetsender.com/>, accessed on: 2021-09-06.
- [74] *Open Network Operating System* (2021), <https://opennetworking.org/onos/>, accessed on: 2021-08-31.
- [75] **Sami, H., Saado, R., Saoudi, A.E., Mourad, A., Otrok, H. and Bentahar, J.** (2023). Opportunistic UAV Deployment for Intelligent On-Demand IoV Service Management, *IEEE Transactions on Network and Service Management*, 1–1.
- [76] **Montemayor, M.M.L., Montillana, J.A., Bautista, A.J. and Zhuo, E.** (2022). Development of Real-Time Remote Sensor Data Transmission System for UAV, *2022 8th International Conference on Control, Automation and Robotics (ICCAR)*, pp.369–374.
- [77] **Supramongkonset, J., Duangsuwan, S. and Promwong, S.** (2021). A WiFi Link Budget Analysis of Drone-based Communication and IoT Ground Sensors, *2021 7th International Conference on Engineering, Applied Sciences and Technology (ICEAST)*, pp.234–237.

- [78] **Sriharsha, C. and Murthy, C.S.R.** (2022). A Novel UAV-aided User Offloading in 5G and Beyond, *2022 IEEE 33rd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pp.1012–1018.
- [79] **Orden, D., Gimenez-Guzman, J.M., Marsa-Maestre, I. and De la Hoz, E.** (2018). Spectrum Graph Coloring and Applications to Wi-Fi Channel Assignment, *Symmetry*, *10*(3), <https://www.mdpi.com/2073-8994/10/3/65>.
- [80] **Athukoralage, D., Guvenc, I., Saad, W. and Bennis, M.** (2016). Regret Based Learning for UAV Assisted LTE-U/WiFi Public Safety Networks, *2016 IEEE Global Communications Conference (GLOBECOM)*, pp.1–7.
- [81] **Xiao, Z., Zhu, L., Liu, Y., Yi, P., Zhang, R., Xia, X.G. and Schober, R.** (2022). A Survey on Millimeter-Wave Beamforming Enabled UAV Communications and Networking, *IEEE Communications Surveys & Tutorials*, *24*(1), 557–610.
- [82] **Cakir, L.V., Huseynov, K., Ak, E. and Canberk, B.** (2022). DTWN: Q-learning-based Transmit Power Control for Digital Twin WiFi Networks, *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, *9*(31), e5, <https://publications.eai.eu/index.php/inis/article/view/1059>.
- [83] **Challita, U., Saad, W. and Bettstetter, C.** (2019). Interference Management for Cellular-Connected UAVs: A Deep Reinforcement Learning Approach, *IEEE Transactions on Wireless Communications*, *18*(4), 2125–2140.
- [84] **Hu, J., Zhang, H., Song, L., Schober, R. and Poor, H.V.** (2020). Cooperative Internet of UAVs: Distributed Trajectory Design by Multi-Agent Deep Reinforcement Learning, *IEEE Transactions on Communications*, *68*(11), 6807–6821.

CURRICULUM VITAE

Name SURNAME: Mehmet Ariman

EDUCATION:

- **B.Sc.:** 2012, Istanbul Technical University, Faculty, Computer Engineering
- **B.Sc. (ERASMUS):** 2011, TU Darmstadt, Informatik
- **M.Sc.:** 2016, Istanbul Technical University, Faculty, Computer Engineering

PROFESSIONAL EXPERIENCE AND REWARDS:

- 2017 - Present Analog Devices Inc.
 - **Technologies:** Firmware, Security, BLE, OTA, I2C, SPI, UART, ISO7816, AES, RSA, ecDSA, TLS, NFC
 - Sr. Manager, SW Systems Design Engineering (May'2022, Current)
 - * Managing mid-size engineering team (10 to 18 people) to support Processor's and Security product lines. Worked as catalyst to setup in-house side-channel-analysis test-bench. Led the hiring activities to grow the team and retain the culture. Also joined product definition efforts for IoT and Security products.
 - * Markets:
 - Industrial
 - Medical Consumable
 - Payment
 - IoT
 - * Skills:
 - Team building
 - Leadership development
 - Performance management
 - Project management
 - Hiring
 - Strategy development
 - Manager, Software(Jan'2021, May'2022)

- * Managing an engineering team (6 to 10 people) to support Processor's and Security product lines. The team supported EMV L2 library development, secure co-processor development, ROM firmware development and IC verification process.
 - * Markets:
 - Industrial
 - Payment
 - IoT
 - * Skills:
 - Team building
 - Performance management
 - Project management
 - Hiring
- Senior MTS, Software (Sep'2018, Jan'2021)
- * Acting as technical lead for BeSpoke software projects developed for external customers. Leading small engineering team. The team developed BLE SCRP device which is demonstrated in TrusTech as first SCRP device. <https://www.stg-maximintegrated.com/en/design/blog/trustech-2018.html> TrusTech Demo
 - * Markets:
 - Payment
 - IoT
 - * Skills:
 - Software Architect
 - Performance management
 - Project management
 - Firmware Development
 - Hiring
- MTS, Software (Jul'2017, Sep'2018)
- * Individual contributor worked on low power and secure payment applications. Contributed to the definition of SDLC flow and integration of CI/CD and static code analysis tools to ensure developer-agnostic repeatable high quality deliveries.
 - * Markets:
 - Payment
 - IoT
 - * Skills:
 - Firmware Development
 - Low-Power
 - Embedded Development

- 2011 - 2017 PAVO Tasarim Uretim Elek. Tic. A.S.
 - **Technologies:** Firmware, Security, OTA, Ethernet, SerDes, UDP, TCP, GMII, RMII, RGMII, SMI, 802.1q, 802.1d, 802.1s, WiFi, SDN, OpenFlow, SIP, Linux, VxWorks, IPC, uBoot, g.SHDSL, I2S, PIM-DM, OpenWRT, RTP, PRP
 - Senior Expert Software Design Engineer (Jan'2017, Jul'2017)
 - * Lead engineer with 4 SW engineers and 1 HW engineer to develop L2/L3/L4 with 10G and 20G ethernet devices and traffic shaper for jitter mitigation.
 - * Markets:
 - Communications
 - Industrial communications
 - * Skills:
 - Project Management
 - Firmware Development
 - Network Protocol Development
 - Embedded Development
 - Mentoring
 - Senior Software Design Engineer (May'2014, Jan'2017)
 - * Senior engineer took responsibility of technically leading wireless networking projects such as wireless access points, SDN access points, industrial ethernet switch and wearable computer.
 - * Markets:
 - Wearable technologies
 - Industrial communications
 - Audio
 - * Skills:
 - Project Management
 - Firmware Development
 - Network Protocol Development
 - Embedded Development
 - Mentoring
 - Software Design Engineer (May'2014, Sep'2011)
 - * Individual contributor worked on driver, protocol and application development for L2/L3 network devices.
 - * Markets:
 - Communications
 - * Skills:
 - Firmware Development
 - Network Protocol Development
 - Embedded Development

- 2021 Maxim Integrated - Leadership Development Program
- 2018 Maxim Integrated - Top Talent Award

PUBLICATIONS, PRESENTATIONS AND PATENTS ON THE THESIS:

- **Ariman M.**, Akkoc M., Sari T.T., Erol M.R., Secinti G. and Canberk B. (2023). Energy-Efficient RL-Based Aerial Network Deployment Testbed for Disaster Areas. *Journal of Communications and Networks*, 25(1), 25-34, 1976-5541 and 1229-2370. (Journal Instance)
- **Ariman, M.**, Cakir, L.V., Ozdem, M. and Canberk, B. (2023). Opportunistic RL-Based WiFi Access for Aerial Sensor Nodes in Smart City Applications, *International Conference on Smart Applications, Communications and Networking (SmartNets) (SmartNets 2023)*, Istanbul, Turkey, p. 5. (Presentation Instance)

OTHER PUBLICATIONS, PRESENTATIONS AND PATENTS:

- Ariman M., Seçinti G., Erel M. and Canberk B. (2015). Software defined wireless network testbed using Raspberry Pi of switches with routing add-on, 2015 IEEE Conference on Network Function Virtualization and Software Defined Network (NFV-SDN), San Francisco, CA, USA, 2015, pp. 20-21. (Presentation Instance)