A Contribution to Communication Management in Private Unmanned Aerial Vehicle Networks

by

Prabhu Jyot Singh

Thesis

Submitted in fulfillment of the requirements for the degree of

Doctor of Philosophy (Business and Informatics)

School of Engineering and Technology

Central Queensland University

Date: 10 February 2020

Candidate's statement

By submitting this thesis for formal examination at CQUniversity Australia, I declare that it meets all requirements as outlined in the Research Higher Degree Theses Policy and Procedure.

Statement of authorship and originality

By submitting this thesis for formal examination at CQUniversity Australia, I declare that all of the research and discussion presented in this thesis is original work performed by the author. No content of this thesis has been submitted or considered either in whole or in part, at any tertiary institute or university for a degree or any other category of award. I also declare that any material presented in this thesis performed by another person or institute has been referenced and listed in the reference section.

Copyright statement

Acknowledgements

The completion of this thesis would not have been possible without the support and encouragement of several special people. Hence, I would like to take this opportunity to show my gratitude to those who have assisted me in a myriad of ways.

I owe my deepest gratitude to my principal supervisor Dr Rohan de Silva. Without his enthusiasm, encouragement, support and endless optimism, this thesis would hardly have been completed. At many stages in the course of this research project, I benefited from his advice, particularly so when exploring new ideas. His positive outlook and confidence in my research inspired me and gave me confidence. His careful editing contributed enormously to the production of this thesis. I express my warmest gratitude to my associate supervisor Dr Indra Seher. I have been fortunate to have supervisors who cared so much about my work, and who responded to my questions and queries so promptly. I cannot think of any better combination of people to have guided me throughout this process. You believed in me when I doubted and, in your ways, encouraged and pushed me with pragmatic advice, feedback and, positivity when I needed it.

The experiments on AR 2.0 were conducted during the development of the LAN of Drones (LoD) project. This project was conducted by my principal supervisor with the approval and financial support from the Deputy Dean (Research), Professor Steven Moore. Sandaruvan Rajasinghege who worked on this project developed much of the LoD network. Without their help and support, I would not have been able to conduct those experiments and I would like to express my heartfelt gratitude to them.

And finally, thank you to my family: my parents, Mr. Amrik Singh and Mrs. Lakhvinder Kaur, deserve special thanks for their continued support and encouragement. Without such a team behind me, I doubt that I would be in this place today. I would especially like to thank my wife, Mrs. Navjot Kaur. She has been extremely supportive of me throughout this entire process and has made countless sacrifices to help me get to this point. The last word goes for Sahaj Preet, my baby boy, who has been the light of my life from the last one year and who has given me the extra strength and motivation to get things done.

Acknowledgement of professional services

Professional editor, Jonathan Dyer, provided copyediting and proofreading services, according to the guidelines laid out in the University-endorsed national guidelines, 'The editing of research theses by professional editors'.

DECLARATION OF CO-AUTHORSHIP AND CO-CONTRIBUTION

1. **Comparison of communication protocols for UAVs and VANETs.**
   Full bibliographic reference: P. J. Singh, R. de Silva, and I. Seher, "Comparison of communication protocols for UAVs and VANETs," in 2016 International Conference on Computing, Communication and Automation (ICCCA), 2016: IEEE, pp. 616-619.
   Status: Published
   Nature of Candidate's Contribution
   I reviewed the existing communication protocols for UAVs and VANETs and wrote the draft version of the paper. My contribution:70%.
   Nature of Co-Authors' Contributions
   The idea was given by Dr Rohan de Silva and both Dr Indra Seher and Dr Rohan de Silva provide suggestions and edited the paper. Contribution by Dr Rohan de Silva: 20%; Contribution by Dr Indra Seher: 10%.
   Has this paper been submitted for an award by another research degree candidate (Co-Author), either at CQUniversity or elsewhere? (if yes, give full details): No
2. **Simulation of inter-UAV communication required for collaborative task execution.**
   Full bibliographic reference: P. J. Singh, R. de Silva, and I. Seher, "Simulation of inter-UAV communication required for collaborative task execution", in 2017 Simulation Australia, [Online]. Available: http://proceedings.simulationaustralasia.com/content/2017/83-paper.pdf.
   Status: Published
   Nature of Candidate's Contribution

I modelled the UAV network in OPNET simulator to test the inter-UAV communication, performed simulations and wrote the draft version. My contribution:80%.

Nature of Co-Authors' Contributions

Dr Rohan de Silva suggested the idea and provided suggestions to improve the draft paper. Both Dr Indra Seher and Dr Rohan de Silva edited the paper. Contribution by Dr Rohan de Silva: 15%; Contribution by Dr Indra Seher: 5%.

Has this paper been submitted for an award by another research degree candidate (Co-Author), either at CQUniversity or elsewhere? (if yes, give full details): No

3. **Design and implementation of an experimental UAV network.**
   Full bibliographic reference: P. J. Singh and R. de Silva, "Design and implementation of an experimental UAV network," in 2018 International Conference on Information and Communications Technology (ICOIACT), 2018: IEEE, pp. 168-173.

   Status: Published

   Nature of Candidate's Contribution

   I constructed the experimental testbed of the private UAV network and wrote the draft version of this paper. My contribution:75%.

   Nature of Co-Authors' Contributions

   Dr Rohan de Silva suggested the idea of the testbed and provided suggestions to improve the draft paper. He edited the paper as well. Contribution of Dr Rohan de Silva: 25%.

   Has this paper been submitted for an award by another research degree candidate (Co-Author), either at CQUniversity or elsewhere? (if yes, give full details): No

4. **Allocation of Control and User Data Channels for Private UAV Networks.**
   Full bibliographic reference: P. J. Singh and R. de Silva, "Allocation of Control and User Data Channels for Private UAV Networks," 2018, Channelization; Video and control traffic; Private UAV networks; UAV communication vol. 7, no. 4.40, p. 5, 2018-12-16 2018, doi: 10.14419/ijet.v7i4.40.24028.

Status: Published

Nature of Candidate's Contribution

I reviewed the existing IEEE 802.11 wireless standards and wrote the draft version of this paper. My contribution:75%.

Nature of Co-Authors' Contributions

Dr Rohan de Silva suggested the idea and provided suggestions to improve the channelization and the draft paper. He edited the paper as well. Contribution of Dr Rohan de Silva: 25%.

Has this paper been submitted for an award by another research degree candidate (Co-Author), either at CQUniversity or elsewhere? (if yes, give full details): No

5. **Cross-Layer Design of Control and Feedback Signal Frames for LANs of Drones.**
   Full bibliographic reference: P. J. Singh and R. de. Silva, "Cross-Layer Design of Control and Feedback Signal Frames for LANs of Drones," 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2019, pp. 447-451.

   Status: Published

   Nature of Candidate's Contribution

   I designed the control and feedback frames for cross layer design of LoDs and wrote the draft version of this paper. My contribution:75%.

   Nature of Co-Authors' Contributions

   Dr Rohan de Silva suggested the idea and provided suggestions to improve the design and the draft paper. He edited the paper as well. Contribution of Dr Rohan de Silva: 25%.

   Has this paper been submitted for an award by another research degree candidate (Co-Author), either at CQUniversity or elsewhere? (if yes, give full details): No

6. **Signaling Protocols for Local Area networks of Drones.**
   Full bibliographic reference: P. J. Singh and R. de Silva, "Signaling Protocols for Local Area networks of Drones", International Journal of Computer Networks & Communications (IJCNC), Vol.12, No.3, May 2020, pp. 99-114, ISSN:09749322.

   Status: Published

   Nature of Candidate's Contribution

I designed the signalling protocols for LoDs and wrote the draft version of this paper. My contribution:75%.

Nature of Co-Authors' Contributions

Dr Rohan de Silva suggested the idea and provided suggestions to improve the design and the draft paper. He edited the paper as well. Contribution of Dr Rohan de Silva: 25%.

Has this paper been submitted for an award by another research degree candidate (Co-Author), either at CQUniversity or elsewhere? (if yes, give full details): No

Abstract

Unmanned Aerial Vehicles (UAVs) or drones have become extremely popular and are used in various commercial applications. They can provide better services for these applications if they work together and form a UAV network. The Flying Ad-hoc Network (FANET), Internet of Drones (IoD) and the private UAV network are such type of UAV networks invented in the past. A private UAV network is beneficial for these applications where all UAVs belong to a person or one organisation. In this research, we implemented the communication between UAVs and UAV to the ground station at the Media Access Control (MAC) layer in the private UAV network by switching through UAV nodes. To test this communication, we implemented two small private UAV testbed networks with Storm 4 Mini and AR 2.0 UAV models. In the Storm 4 Mini private UAV network, the video signals were transmitted over the 5GHz network while each UAV was controlled through the analog signals from their remote controller over the 2.4 GHz frequency spectrum. Whereas, control and video signals were transmitted over the 2.4 GHz network in the wireless frame format for the AR 2.0 UAV testbed network. This research outlines the real-time practical problems for communication between UAVs and UAV to the ground station in the private UAV network and provides the solutions for those problems. The video transmission delay from each UAV to the ground station was one of them. This video delay was presented in the network due to the use of a common single communication channel between UAVs. To resolve this issue, we proposed and developed new data and control channels that can be used for the communication between UAVs and UAV to the ground station in this UAV network. This new channelisation avoids congestion and packet drop for video and control signal traffic in this network. In a private UAV network where UAVs are connected in tandem, the UAVs that are in the middle of the network have to carry the control and video signals of other UAVs. Given the limited processing power and dynamic memory capacity of UAVs, this would increase the queuing delays for transmission these signals. As such, we studied the frame formats of existing control, feedback and video signals for the smartphone operated commercial UAV and proposed a new approach of signaling for control and feedback frames for the private UAV network. The control and feedback signals transmission delays from each UAV to the ground station were calculated with both types of signaling mechanisms for this UAV network. The result of these delays comparison shows that newly developed signaling mechanism for the private UAV network with the single control and feedback frame has less delay on the average.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AODV	Ad-hoc On-Demand Distance Vector
AP	Access Point
ATC	Air Traffic Control
CAR	Connectivity Aware Routing
CBF	Contention Based Forwarding
DCR	Data Centric Routing
DHCP	Dynamic Host Configuration Protocol
DSDV	Destination Sequenced Distance Vector
DSR	Dynamic Source Routing
DSRC	Dedicated Short Range Communication
DTN	Delay Tolerant Network
DYMO	Dynamic MANET On-demand
EWM	Emergency Warning Message
FANET	Flying Ad-Hoc Network
FCS	Frame Check Sequence
FPV	First Person View
FTP	File Transfer Protocol
GPS	Global Positioning System
GPSR	Greedy Perimeter Stateless Routing
GRANT	Greedy Routing with Abstract Neighbour Table
GyTAR	Greedy Traffic-Aware Routing
HD	High Definition

HDMI	High Definition Multimedia Interface
IMC	Inter-module Communication
IMU	Inertial Measurement Unit
IoD	Internet of Drone
IP	Internet Protocol
ISDN	Integrated Service Digital Network
LAN	Local Area Network
LLC	Logical Link Control
MAC	Media Access Control
MANET	Mobile Ad-Hoc Network
MIMO	Multiple Input and Multiple Output
NDTN	Non-Delay Tolerant Network
OLSR	Optimised Link State Routing
OPNET	Optimised Network Engineering Tool
OSI	Open System Interconnection
PD	Processing Delay
PHY	Physical
PRD	Propagation Delay
QAM	Quadrature Amplitude Modulation
QD	Queuing Delay
RERR	Request Error
RF	Radio Frequency
ROV	Remotely Operate Vehicle

RREP	Route Reply
RREQ	Route request
RSU	Road Side Unit
SDK	Software Development Kit
SSID	Service Set Identifier
STP	Spanning Tree Protocol
TCP	Transmission Control Protocol
TD	Total Delay
TORA	Temporally Ordered Routing Algorithm
TRD	Transmission Delay
UAS	Unmanned Aerial System
UAV	Unmanned Aerial vehicle
UDP	User Datagram Protocol
USMP	UAV Search Mission Protocol
VANET	Vehicular Ad-Hoc Network
WAP	Wireless Access Point
WAVE	Wireless Access in Vehicular Environment
WDS	Wireless Distribution System
WISP	Wireless Internet Service Provider
WSMP	Wireless Short Messaging Protocol

List of Publications

Journal articles

1. P. J. Singh and R. de Silva, "Allocation of Control and User Data Channels for Private UAV Networks," 2018, Channelization; Video and control traffic; Private UAV networks; UAV communication vol. 7, no. 4.40, p. 5, 2018-12-16 2018, doi: 10.14419/ijet.v7i4.40.24028.
2. P. J. Singh and R. de Silva, “Signaling Protocols for Local Area networks of Drones”, International Journal of Computer Networks & Communications (IJCNC) Vol.12, No.3, May 2020, pp. 99-114, ISSN:09749322.

Conference papers

3. P. J. Singh, R. de Silva, and I. Seher, "Comparison of communication protocols for UAVs and VANETs," in 2016 International Conference on Computing, Communication and Automation (ICCCA), 2016: IEEE, pp. 616-619.
4. P. J. Singh, R. de Silva, and I. Seher, “Simulation of inter-UAV communication required for collaborative task execution”, in 2017 Simulation Australia, [Online]. Available: http://proceedings.simulationaustralasia.com/content/2017/83-paper.pdf.
5. P. J. Singh and R. de Silva, "Design and implementation of an experimental UAV network," in 2018 International Conference on Information and Communications Technology (ICOIACT), 2018: IEEE, pp. 168-173.
6. P. J. Singh and R. de. Silva, "Cross-Layer Design of Control and Feedback Signal Frames for LANs of Drones," 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 2019, pp. 447-451.

Chapter 1

Introduction

1.1 Introduction

A UAV, commonly known as a drone, is a flying object capable of covering large distances to capture aerial images and videos. UAVs that are currently available in the market can be differentiated based on size such as nano, micro and mini UAV, where the small size micro UAV is extremely useful across various commercial applications, and are being tested for use in as diverse areas as farming, surveillance, parcel delivery, rescue, project management and natural disaster relief [1], [2], [3], [4], [5]. This micro UAV also has been reported as a lifesaving tool at the beaches. In 2018, Surf Life Saving NSW (SLSNSW) used a micro-size UAV to provide lifesaving vests to two swimmers, consequently saving their lives [6], [7]. Additionally, the applications of UAVs can change the way small businesses and individuals conduct their businesses, potentially saving both time and money. For example, real estate agents can now use this micro UAV to live stream a video of a property to a potential buyer without them having to visit the property. So, they can save time without going to that property location and also save money as they do not need to hire more people to do this job. Also, farmers can now use the live streaming feature of micro UAV to monitor their farms from a single location. These are just a few examples of commercial application of micro UAVs that are currently available in the market.

A micro UAV is able to send the live video stream to the ground station or its controller with the help of various components. As such, a small micro UAV is designed to operate with minimal hardware and software components: a mechanical UAV object; a ground control station, which can be a radio frequency controller, a mobile phone or tablet; onboard operating system; and, finally, a communication system to connect and exchange the messages between onboard UAV and the ground station. Different communication technologies are used by the UAV manufacturer to connect the ground station with the onboard UAV system. The analog radio frequency signal is used to connect the UAV to the ground station in the earlier UAV models [8], [9], [10]. However, the IEEE 802.11 Wi-Fi standards are now being used in most of the UAV models to connect the UAV to its ground station (such as smartphone or tablet) —the AR 2.0, 3DR Solo, and an advanced model of the

DJI and Parrot UAVs are some examples of Wi-Fi-controlled UAVs [8], [10], [11]. A micro UAV provides various services to its respective ground stations. It can fly to specific locations, sending live video of far-off areas to its ground station, and videos and images can also be recorded on the onboard memory (memory card) of the UAV. Additionally, this micro UAV has some payload capacity (that is, the weight a UAV can carry), which can be used to deliver objects from one location to another. In the majority of UAV applications, however, the live video streaming at the ground station is the predominant function. These UAVs can provide more sophisticated services for small businesses and individuals in the various commercial applications domain (such as, farming, surveillance, parcel delivery, rescue, project management and natural disaster relief) if they work together by forming a network.

Flying Ad-hoc Network (FANET) is a UAV network proposed by Bekmezci, et al. [12] in 2013. The FANET is similar to the Mobile Ad-Hoc Network (MANET) and the Vehicular Ad-Hoc Network (VANET), but nodes in the FANET are in the form of flying objects. Gharibi, et al. [13] proposed a UAV network, Internet of Drones (IoDs) in 2016. IoDs is a special type of UAV network with layered architecture designed to coordinate UAV access. A FANET comprises high node density and these UAV nodes in this network belong to different users. It is not suitable for commercial UAV applications where the number of UAVs is comparatively small, and all UAVs belong to the same owner or one organisation. The implementation of IoDs network requires multiple base stations, and that is impractical for the commercial UAV applications. The idea of the private UAV network was proposed by de Silva [14] in 2013 for the use of commercial applications of UAVs. A private UAV network is the network of UAVs that have minimal number of UAVs belonging to the same owner or one organisation. The performance of this private UAV network depends on its communication system (the communication between UAVs and UAV to the ground station). As such, it is required to implement a better approach of communication between UAVs and UAV to the ground station for this UAV network. The real-time practical problems for communication between UAVs and UAV to the ground station also needs to be investigated for better performance of this network. Therefore, through this research, the communication between UAVs and UAV to the ground station in the private UAV network is implemented and tested by forming private UAV testbed networks.

1.2 Rationale and significance of the study

Each UAV in a private UAV has to send its data directly to the ground station or through the intermediate UAVs. This data need to temporarily stored at each UAV along the path of data flow and then forwarded to the next UAV. Furthermore, each UAV also adds its own data before it transmits the data from previous UAVs to the next UAV.

Most of the micro UAVs that are available in the market use IEEE 802.11 b/g/n standards for the communication between the UAV and the ground station [8], [10], [11]. A private UAV network comprising of these micro UAVs will also need to use the same IEEE 802.11 standards to transmit the control, feedback and data signals between UAVs and the ground station. When there are many UAVs in a private UAV network and real-time traffic of good quality such as live video has to be transmitted from each UAV to the ground station, the node delays may deteriorate the quality of video and can have a significant effect on the timely arrival of control and feedback signal packets at the intended destination. This may lead to poor control of the UAVs of the network.

There are several routing techniques used in other types of UAV networks. An interesting issue to consider is if we could borrow a routing technique from them or design a suitable other communication method considering the special topology of private UAV networks. As, the communication possibilities between UAVs and UAV to the ground station in a private UAV network needs to be explored and an efficient approach has to be designed.

1.3 Outline of the thesis

The thesis is organised as follows. Following the introduction, Chapter 2 provides a comprehensive background of the study. In this chapter, we discuss the different UAV types currently available in the market, as well as the use of UAVs for various commercial applications. This chapter also contains a literature review on the VANET and UAV communication protocols. In Chapter 3, we discuss the private UAV network and test the communication between UAVs and UAV to ground station through OPNET simulator. This chapter also deals with the discussion of communication between UAVs and UAV to the ground station at MAC layer for the private UAV network by switching through the UAV nodes. To test these communications, two private UAV testbed networks were used in this research. We present an outline of the practical setup for the private UAV networks with six UAVs in terms of video transmission from one UAV to another and show that all the videos from the six UAVs are receiving at one ground station. All the details regarding the configuration of the network setup and devices are given in this chapter. Furthermore, we performed various test cases to verify the communication between UAVs and UAV to ground station in both testbed networks by switching through UAV nodes and these are also presented in Chapter 3. Finally, Chapter 3 discusses the challenges faced during the communication between UAVs and UAV to the ground station in these private UAV networks. In these testbed private UAV network experiments, a live video streaming delay was experienced during the transmission of the video data from each UAV to the ground station, and this led to the development of new control and data channels for the network. This new channelisation scheme for the private UAV network is outlined in Chapter 4 of the thesis. For this channelisation, we analysed existing data channels for the 2.4 and 5 GHz spectrums and then developed new control and data channels based on the needs of the private UAV network. In Chapter 5, an overview of the management of signaling protocols for the private UAV network is given. This chapter also provides a discussion on the current control and feedback signaling mechanism for private UAV. Accordingly, we proposed a new signaling mechanism capable of reducing the signal transmission delay from each UAV to the ground station in private UAV network, and the discussion about the new signaling mechanism in this network using a single control and feedback frame is given in this Chapter. Finally, Chapter 6 concludes the thesis with recommendations for future research in the field. Appendices are also included to provide additional information regarding the commercial UAVs currently available in the market, as well as information regarding AR 2.0 UAV’s control and feedback commands.

Chapter 2

Background

2.1 Introduction

As discussed in Chapter 1, micro UAVs are very useful in the various civilian applications, and they can provide more efficient services by forming a network. Furthermore, a private UAV network proposed in past research is the best suitable network for these applications and can be used by an individual or one organisation according to their need. The performance of this network can be improved by providing a better communication system. Therefore, it is required to study the various types of UAVs that are currently available in the market and their applications in the commercial domain to implement communication in the private UAV network. It is also essential to study all types of UAV networks (FANETs, IoDs and private UAV network) as well as ground-based wireless ad-hoc networks (MANETs and VANETs) and the communication protocols used in these networks for the implementation of the better communication system in the private UAV network. This chapter covers a detailed discussion about these networks and their communication protocols. As discussed in the introduction chapter, two experimental testbed networks were implemented in this research to test the communication between UAVs and UAV to the ground station. As such, the basic functionality and capability of the UAV modes, that were used in the experimental testbed networks are also discussed in this chapter.

The remainder of the chapter is as follows. Sections 2.2 and 2.3 introduce UAVs and their different types respectively. Section 2.4 discusses the application of UAVs, with a focus on their commercial use (for example, in site monitoring, agriculture, and surveillance). Section 2.5 outlines MANET and VANET, both of which have some common characteristics as per the wireless ad-hoc UAV network. This Section also includes the discussion about the existing UAV networks (that is, FANET, IoDs and private UAV network) and challenges associated with each UAV network as well as adaptability of these UAV networks for the commercial applications of UAVs. The wireless ad-hoc routing communication protocols are also discussed in this Section. The capabilities and functionality of Storm 4 Mini and the AR 2.0 UAV are presented in Section 2.6. Finally, a chapter summary is given in Section 2.7.

2.2 UAVs

UAVs, commonly known as drones, are capable of flying autonomously to a specified location along a determined path and can also be remotely controlled by a human operator. These unmanned vehicles are also referred to as Unmanned Aerial Systems (UASs) and Remote Pilot Aerial Systems (RPASs). The US Department of Defense defines a UAV as an unmanned vehicle that can fly autonomously, carry some weight and remotely operated. As such, the cruise missiles and artillery projectile cannot be considered as UAV [15].

UAVs were initially used for military purposes; nowadays, however, their popularity and functionality have extended to use in commercial environments. In recent years, UAV technology has received considerable attention from many researchers due to the numerous application possibilities associated with it. At the time of writing, the number of operating UAVs is limited; however, a dramatic increase in their number and service applications (for example, in disaster management, agriculture, construction, surveillance, and transportation) is predicted for the foreseeable future [16].

2.3 Types of UAVs

At present, various types of UAV are available in the market. A study by Hassanalian and Abdelkefi [17] classified the different UAV types based on their functionality, altitude and distance. Table 2.1 presents different UAVs based on their functionality. As such, UAVs can be used for target and decoy, reconnaissance, combat, logistics and research and development. UAVs can also be classified based on their payload (weight a UAV can carry), flying capacity and altitude. Table 2.2 presents the different UAV types with respect to their altitude and distance. As we can see from Table 2.2, size is an important factor to differential the UAVs. The micro UAVs are used in most of the commercial applications that are discussed in the next subsection. To test the communication between UAVs in the private UAV network, these micro type UAVs are being used.

Table 2.1 Classification of UAVs based on their functionality

UAV types	Category	Uses
1	Target and decoy	target enemy aircraft
2	Reconnaissance	battlefield intelligence
3	Combat	high-risk missions
4	Logistics	cargo operations
5	Research and development	further research on UAVs
6	Civil and commercial UAVs	commercial applications

Table 2.2 Classification of UAVs based on their altitude and distance

UAV types	UAVs	Altitude	Distance	Weight range
1	Pico Air Vehicle (PAV)	Few meters	Very less range	< 100 mg
2	Nano-Air Vehicle (NAV)	Up to 200 m	Less range but better than PAV	< 100 g
3	Micro UAV	Up to 1500 m	Cover 2 km	<10 kg
4	Tactical	Up to 5500 m	Cover 160 km	Heavyweight UAVs
5	MALE	Operate at 9000 m	Indefinite field	Heavyweight UAVs

2.4 Applications of UAVs

2.4.1 UAVs for disaster prediction and recovery

Natural disasters have a significant impact on human life. Common examples of natural disasters include (among others) earthquakes, floods and bushfires. UAV research groups have worked on disaster management from different perspectives. For example, the remote sensing application of micro-size UAV helps to monitor and evaluate flood disasters by collecting real-time data and generating a flood path to prevent further flood damage. A micro-size UAV is also helpful in meteorological disaster monitoring, invention, and prevention of landslides and mud-rock flow. However, the main application of UAV in the natural disaster is the disaster relief and evaluation of damage associated with earthquakes and their surrounding areas. For this application, a UAV rapidly collects images of an earthquake's surrounding area, in turn helping to determine the extent of the earthquake strike, as well as the distress severity to nearby buildings, structures and engineering facilities in need of rapid repair—this offers a basis for decision-makers to organise and dispatch disaster relief [2].

Bushfire is another area where disaster prevention using UAV has become popular. Specifically, a UAV is used to monitor bushfires by giving regular updates via images and videos of the extent of the fire damage. Similarly, the New South Wales Fire Brigade uses a micro-size UAV to monitor bushfires [7]. Furthermore, a UAV is used to collect real-time data for post-disaster management [18]. Another application of a micro UAV in the natural disaster is the flood path prediction in desert environments using a Lagrangian (mobile) microsensor. This sensor passes a signal to the UAV after that real-time data is sent to a ground station to provide the most up-to-date information on the disaster zone [19]. Tuna, et al. [20] conducted a study on the communication between a UAV and a ground station, revealing that a UAV is extremely helpful in real-time data processing for disaster management.

Overall, various applications for micro UAV have been proposed and have proven helpful in disaster management efforts; however, an investigation into UAV applications for disaster prediction is limited.

2.4.2 UAVs for agriculture and farming

UAVs are used extensively in the agriculture application domain. In France, Lelong, et al. [21] conducted a study on wheat crop management with a UAV. In this study, images of crops were captured by a UAV during the growing season. These images were later used to develop a vegetation index, and the results were compared with ground-measure parameters. This study demonstrated the data generated with a UAV was more stable and more accurate as compared to ground-measure parameters.

Likewise, a UAV-based system (VIPtero) was designed for vineyard management (that is, precision agriculture). The VIPtero is a flexible system and capable of taking high-resolution images of agriculture farms. This study produced strong findings regarding comparisons between the vegetation index generated using UAV data and a ground-based spectrometer [5].

UAVs were also used to spray pest and weed control chemicals on growing crops. Costa, et al. [22] described how a UAV is helpful for the calculation of precise pesticide quantities needed for farm application. Specifically, a wireless sensor device was used in this study to generate feedback on how much chemicals were required for a given area. A micro-size UAV was used in this study to spray the chemical over the given area based on the feedback generated by the wireless sensors. This simulation-based study also supported the use of a UAV in pesticide distribution for windy conditions. Furthermore, Chelard [23] study focused on different UAV types commonly used in agriculture for the capturing of video and image data.

2.4.3 UAVs for surveillance

UAVs are also very useful for surveillance purposes. For surveillance application, a camera is set up on a UAV which monitors a particular object and then sends the object's video stream to a ground station. The motion of an object is also detected by UAVs, which helps to identify specific objects more accurately. Likewise, Quigley, et al. [24] study supported the use of UAVs in the detection, identification, and location of a target. For this practical approach, gimbal hardware, a flight-path generation algorithm, and a human UAV-interaction scheme were employed.

The use of a UAV may also provide a more economical approach to traffic monitoring by relevant authorities, with surveillance allowing for data on traffic volumes in real-time. Chen, et al. [3] proposed an idea for data linkage between a mounted UAV camera and a monitoring terminal located on the ground. In this approach, video captured by a UAV is first transferred to a ground station and then to a traffic control office. Such UAV surveillance system was capable of delivering good quality real-time video to traffic control offices.

Similarly, a UAV surveillance system was developed by Semsch, et al. [25]. This system was an autonomous UAV surveillance system which could be used in complex urban environments. A UAV-based system, Moving Object Detection and Tracing (MODAT) was developed in 2010. This system was used to send a live video stream with the detection of moving objects to a ground station [26].

2.4.4 UAVs in the construction industry

UAVs have also become popular in the construction industry. Accordingly, archaeologists have used UAVs for site excavation. Rinaudo, et al. [27] conducted a study with a UAV for the management of excavation at a site which guaranteed the production of orthophotos (an aerial photograph that has been geometrically corrected). In this study, the Hexacopter UAV was used for site acquisition, and triangular aerial images were generated and then converted into solid orthophotos. Compared to more traditional approaches used in site excavation, the UAV produced more promising results in site mapping [27].

UAVs are also used to display an aerial view of a completed project. As safety is a primary concern on construction sites, a UAV can be used to check whether safety practices are duly adopted while construction is undertaken. Similarly, construction companies have used UAVs to create 3D maps by capturing the aerial images of their construction sites [28].

2.4.5 Other UAV applications

In addition to the above mentioned applications, micro size UAV has also been tested for parcel delivery in Germany by Deutsche Bundespost [4]. A UAV also allows the news reporters to take aerial videos of incidents to be covered in their reports. Additionally, UAVs are a powerful tool in real estate. For example, a client can view live aerial photography of a property with the help of the UAV before purchasing it [29].

2.5 Wireless ad-hoc networks and UAV networks

In Section 2.4, the various categories of UAVs along with their respective applications of micro UAVs, are discussed. Although the use of a single UAV in the commercial space is effective, the use of more than one UAV to form a UAV network is becoming increasingly common. UAV networks are defined in the past literature known as FANET, IoDs and private UAV network. Furthermore, FANET is wireless ad-hoc networks that have some similar characteristics with ground-based ad-hoc networks such as MANET and VANET. The functionalities, challenges and limitations associated with these networks are addressed in the existing literature. In this section, we discuss all of these networks and the communication protocols that were developed for these networks.

2.5.1 Mobile ad-hoc network (MANET)

A MANET is a type of wireless ad-hoc network. In this network, mobile devices are connected via a wireless medium. This network is self-configurable, require less infrastructure and can act as a replacement for the traditionally more expensive network. In the MANET, any two mobile devices which are within communication range are capable of sending signals to each other. MANET has become popular due to its cost-efficient mechanisms. Each mobile node in a MANET works as a router and, therefore, no separate router is needed for signal transmission from one mobile device to others. Furthermore, MANET serves as a base network for VANET and FANET by sharing some common characteristics of these networks. A mobile device in the MANET is replaced by a moving vehicle and by a flying object in VANET and FANET, respectively. However, each network has its own capabilities and associated challenges. For example, in VANET, the nodes move rapidly, but they are located on the ground and, in the case of FANET, the nodes are in the sky with high mobility, making both networks more challenging as compared to the MANET [30].

2.5.2 Vehicle ad-hoc network (VANET)

VANET is considered as a sub-class network of MANET. In VANET, moving vehicles communicate with each other through wireless technology. The high-speed of moving vehicles is a key challenge in designing communication protocols for VANET. In such communication, each vehicle is treated as a wireless router and, thus, forwards its data packet to another vehicle that is in communication range. VANET uses wireless ad-hoc routing protocols to exchange information between vehicles, as well as between vehicles and Road Side Units (RSUs). In the VANET, basically three types of communications happens; first the communication inside the vehicle using an application unit known as in-vehicle domain;

second Vehicle to Vehicle (V2V) and Vehicle to roadside Infrastructure (V2I) communication that is called a the ad-hoc domain; and third known as an infrastructural domain, where communication occurs between the vehicle and the Internet. The wireless technologies used by VANET comprise the cellular system, WLAN/Wi-Fi, WiMAX, Dedicated Short Range Communication (DSRC) Wireless Access in Vehicular Environments (WAVE) and the combined wireless access technologies [31].

The most commonly used technologies for V2V communication are IEEE 802.11, DSRC and the General Packet Radio Services (GPRS). The main challenges associated with VANET communication are the scalability of protocols, security and high-speed real-time communication [32]. An example of simple VANET architecture is presented in Fig. 2.1. The V2V communication in VANET is represented by the green dotted line in Fig. 2.1. The vehicles in VANET are also connected with an RSU, and the blue dotted line in Fig. 2.1 represents this V2I communication. The third type of communication is inter-roadside communication; it is the communication between two RSUs and shown in the red dotted line in Fig. 2.1. This network has some common characteristics with MANET; however, its functionality differs from a traditional MANET network as the fast-moving vehicles, communication protocols, battery power and communication reliability remain the main challenges associated with this network.

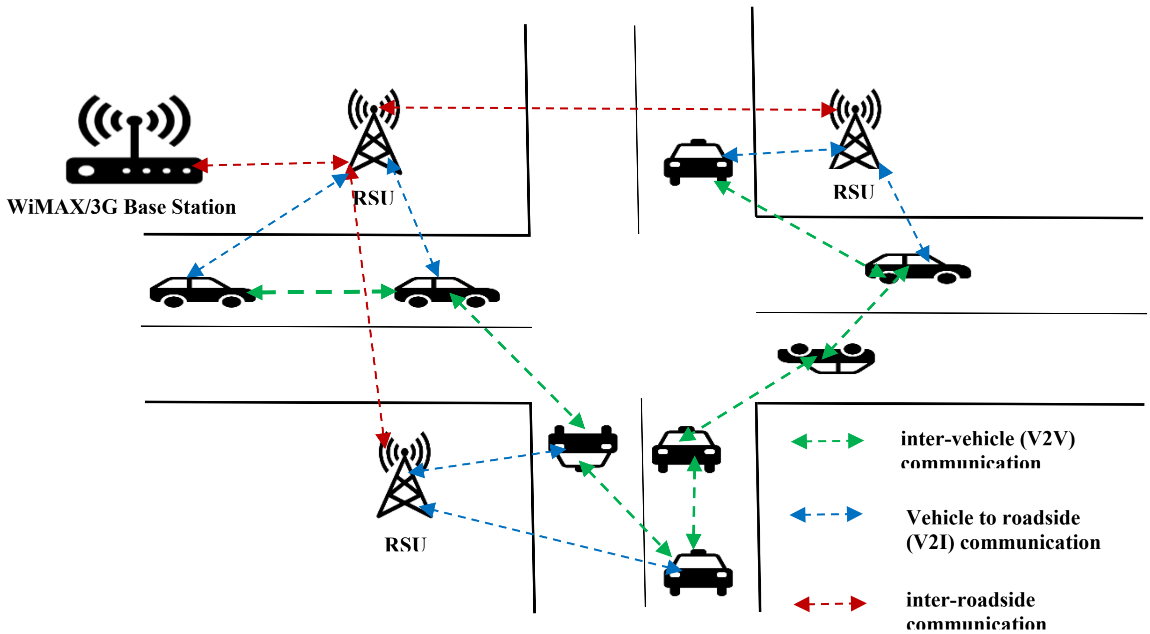


Fig. 2.1 VANET architecture

In most of the intelligent transport system application, the WAVE protocols are used to provide direct connectivity between vehicles through DSRC. In the following sections, brief information about the WAVE protocol stack is presented.

2.5.2.1 WAVE protocol stack

The VANET uses the communication protocols of the WAVE protocol stack for communication between two vehicles and between a vehicle and RSU. This protocol stack contains different layers and protocols, similar to the OSI model, and it is presented in Fig. 2.2.

<table>
<tr><td>Non-safety Applications</td><td>Safety Applications
SAE J2735</td></tr>
<tr><td>Transport Layer
TCP/UDP</td><td rowspan="2">Wireless Short Messaging Protocols (WSMP)
IEEE 1609.2 (Security)
IEEE 1609.3</td></tr>
<tr><td>Network Layer
IPv6</td></tr>
<tr><td colspan="2">Logical Link Control (LLC)
IEEE 802.2</td></tr>
<tr><td colspan="2">Media Access Control (MAC) Layer
IEEE 802.11 P
IEEE 1609.4</td></tr>
<tr><td colspan="2">Physical Layer
IEEE 802.11 P</td></tr>
</table>

Fig. 2.2 WAVE protocol stack

As shown in Fig. 2.2, the first layer of this stack is a physical layer that uses IEEE 802.11 P standard protocols. Similar to the data link layer of the OSI model, this stack has MAC and LLC layer. The MAC layer uses IEEE 802.11 P and IEEE 1609.4 protocols, and the LLC layer uses IEEE 802.2 protocols. The WAVE protocols stack has the network and transport layer protocols similar to the other networking model (that is, OSI, TCP/IP), but it has additional WSMP protocols as shown in Fig. 2.2. The top layer of this stack defines the protocols for the non-safety and safety application [33].

2.5.3 Flying ad-hoc network (FANET)

A FANET is a wireless ad-hoc network or more precisely a mobile ad-hoc network in which the flying objects (for example, UAVs) communicate with each other by forming a network. Compared to the MANET and VANET, this is a more challenging network due to the high mobility of UAVs located in the sky. The FANET is formed with the number of UAVs that belong to the different users and are capable of communicating with each other in a long-range network. In the FANET, each UAV must be able to sense the position of the other UAVs to avoid a collision. Another design challenge of this network concerns the node density. The node density of the FANET is very low compared with the MANET and VANET, as the UAVs are scattered in the sky. Furthermore, compared with the MANET and VANET, the topology in the FANET changes more frequently. Notwithstanding the abovementioned challenges, the power and energy consumption associated with the FANET form the most significant challenge for the use of this network. As each UAV flies in the sky, they consume battery power and, if these UAVs carry a payload, then battery time is reduced even further. Therefore, the biggest concern in this network is recharging each UAV battery while they continue operating.

Additionally, the communication in FANET is also a key challenge, as the distribution of UAVs in the FANET can be within the same plane, or they can communicate at different altitudes (a simple FANET is presented in Fig. 2.3). In the FANET, five different types of communication occur: 1) inter-plane communication, where communication between all UAVs occurs in the same plane; 2) intra-plane communication, where a number of UAVs on one plane communicate with those on another plane; 3) UAVs communicate with a ground station; 4) ground sensor communication, each UAV in the FANET contains many sensors and, as such, ground sensor communication is also considered a type of communication; and, finally, 5) FANET-VANET communication, where a FANET in the sky communicates with a ground VANET network. All six of the abovementioned communication types are illustrated in Fig. 2.3.

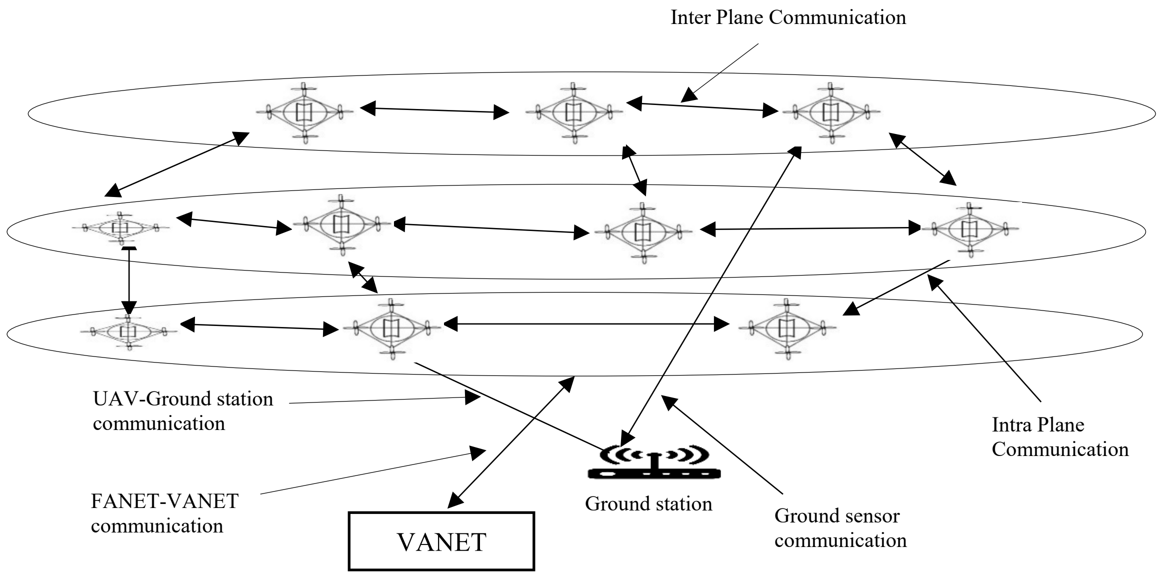


Fig. 2.3 FANET architecture

Different network topologies are being used to connect UAVs in FANETs. For example, a star topology is used in FANET, in which a ground station is connected with one UAV, which is further connected with other UAVs to form a single branch of this network. A FANET can have many of these branches connected through the star topology. Other types of topology include the multi-star topology and the mesh typology, both of which can be used to communicate with one UAV to another within the FANET [12].

2.5.4 Internet of drones (IoDs)

An IoDs is a UAV layered network control architecture and was designed to provide coordinated access and navigation services of UAVs to controlled airspace. The IoDs network shares some similarities with three technological innovations (that is, an Air Traffic Control (ATC) system, cellular network and the Internet). The IoDs network is based on an ATC system, which is used to orchestrate the collision-free movement of all flights in the air space. For this purpose, controlled coordination of UAVs in the air space is required and is the primary concern in the IoDs architectures. This UAV network also shares some similarities with a cellular network, which is used to send data and signals from one mobile device to another mobile device. Similar to the cellular network, IoDs requires multiple base stations that are connected through a wired or wireless link. Finally, the most important technological innovation is the Internet, which also shares similarities with the IoDs in that both use routing to send information from one node to another [13].

2.5.5 Private UAV network

The idea of the private UAV network was proposed by de Silva [14] in 2013 to use it in the commercial applications of UAVs where all UAVs belong to the same owner or one organisation. A FANET cannot be used in such commercial applications of UAVs as it has comparatively high node density and UAVs in this network belong to the different user. On the other hand, an IoDs network is based on the Internet, as well as on a cellular system and communicates through many base stations that are required to place in the same geographical area of the network; thus, for the adaptability of a UAV network in the commercial UAV applications, the IoDs network is also not appropriate for small organisations or individuals. After considering these issues in FANET and IoDs, the idea of private UAV network was proposed for these commercial applications.

A private UAV network is a network of UAVs in which all the UAVs belong to the same owner or one organisation. In this network, all the UAVs are controlled by a single ground station, and each of them sends live video streaming to that ground station. A simple private UAV network with three UAVs is shown in Fig. 2.4. The communication in this private UAV network is accomplished by passing the control and data signals from one UAV to another. As such the UAV 3 transmit its data to UAV 2, UAV 2 transmits the data form UAV 3 as well as its own data to UAV 1 and UAV 1 transmit data of all UAVs to the ground station. Similar to this the ground station also controls all the UAVs by transmitting their control commands for all the UAVs in this network.

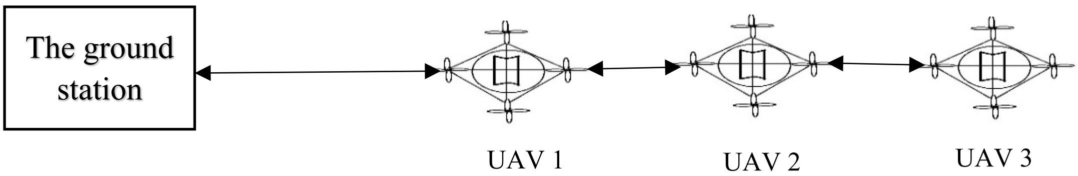


Fig. 2.4 A private UAV network with three UAVs

2.5.6 Wireless ad-hoc routing protocols for UAV networks

A UAV network relies upon the coordination between the UAVs, and it is achieved through the use of proper communication protocols, UAV routing and pathfinding. As such, communication protocols play an essential role in the UAV network—they are its backbone. There are various ways to classify existing wireless ad-hoc routing protocols based on message forwarding or with the routing table. In the following subsections, the details of these protocols are given with the discussion of their adaptability for the UAV networks.

2.5.6.1 Static routing protocols

The communication in a wireless ad-hoc network through static routing protocols is accomplished through the static routing table. During the communication between nodes in a network, this table cannot be updated or changed. Each node in the network that is using the static routing protocols communicates with its neighbour nodes according to the pre-calculated static path in the routing table. Therefore, these protocols are not suitable for the UAV network where the UAV position changes dynamically, and their use in the UAV network is minimal. Some of the static routing protocols proposed in past research are discussed in the following subsection.

Load Carry and Delivery Routing (LCDR)

As the name shows the LCDR static routing protocol works on the pre-calculated static path that is stored in the static routing table. In this protocol, a ground station node transmits its data to the far end node by using a single UAV. This UAV carries the data and transmit it to the destination node. This protocol maximises the throughput but faces a significant data delivery delay due to the use of a single UAV to deliver the data at the destination node. The throughput can be increased in this protocol by sending multiple UAVs to relay the information to various destinations [34].

Multiple Level Hierarchical Routing (MLHR)

MLHR is a static routing protocol which works on the hierarchical structure of the network rather than the flat structure that is used in VANETs. In the hierarchical structure of the network, the UAVs are grouped into a cluster for the communication from one UAV network to the other. Furthermore, a cluster head can have a connection outside of its cluster, and a node can send its data inside the cluster by broadcasting its data. This protocol generates a significant overhead in the network due to the rapid changing of the cluster in the UAV network [35].

Data Centric Routing (DCR)

The DCR approach is based on the data content when many nodes request the data; as such, this routing can be used from one to many data transmissions within the network. In this routing algorithm, the consumer and producer nodes are used. The consumer node can be the UAV-ground-station or the subscription message for collecting data from a specific area, and the producer node is used to decide data dissemination. Once the UAV receives the data, it checks it according to the subscription message and then forwards it to another node in the network [12].

2.5.6.2 Point-to-Point Routing Protocols

Point-to-point routing protocols for wireless ad-hoc networks are used to transmit data from one node to another and subdivided into two categories: the topology base and the position base (as shown in Fig. 2.5).

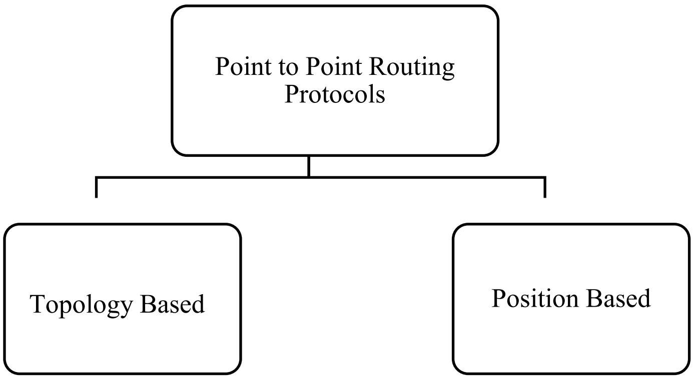


Fig. 2.5 Point-to-point routing protocols

Topology Based Routing Protocols

The topology of a network determines the interconnectivity of nodes in a network. In these protocols, the data packet is forwarded from one node to another with the help of link information. These protocols are further subdivided into two categories: proactive and reactive routing protocols. The routing information in the proactive routing protocols is periodically stored in the routing table. On the other hand, in reactive routing (on-demand routing), the routing table is maintained only when the routing path is required. A summary of the topology based routing protocols is given in Fig. 2.6.

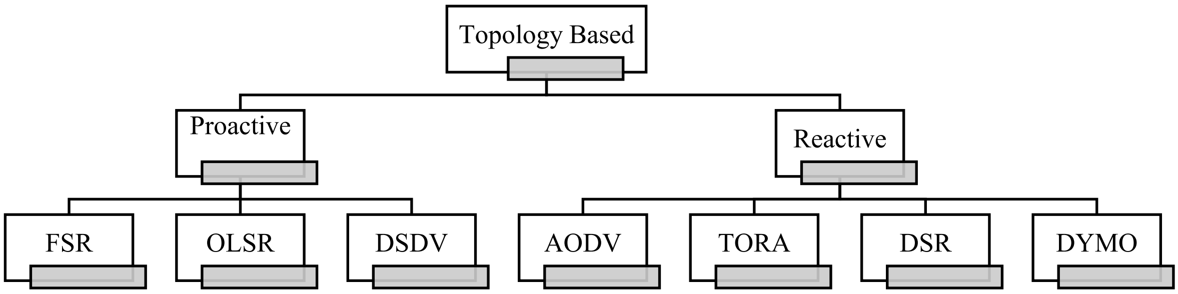


Fig. 2.6 Topology based routing protocols

Fisheye State Routing (FSR): This routing protocol is used for the communication in the ad-hoc wireless network and works as a proactive routing protocol. In this protocol, each node maintains a topology table which is exchanged between local neighbours. The frequently small update of this table mostly occurs between local neighbours rather than between remote neighbours during the communication between nodes in the network. In this routing protocol, each node contains information about the next hop routing table, neighbour list, topology table and distance table [36].

In a UAV network, where all nodes are mobile nodes, and they are moving very fast, the updating of the routing table may introduce some inaccuracy.

Optimised Link State Routing (OLSR): As name shows, the link state routing algorithm is used in OLSR routing protocol (proactive routing). This routing protocol works well for the network that has dynamic topology. In this protocol, each node stores a routing table with optimal path information. A data packet is transferred from one node to another in the network through this optimum path. Due to the multipoint relay node, this protocol utilises the total available bandwidth. Furthermore, OLSR takes more time to find an alternative path when the communication link between nodes is broken [37].

The position of the UAV is changed dynamically, and very fast in the UAV network, this would lead to transmitting of more control commands between UAVs. Therefore, the communication between UAVs with this routing protocol faces more control message overhead and packet loss in the UAV network.

Destination Sequenced Distance Vector (DSDV) Routing: DSDV routing is a routing protocol, used for communication between nodes in a wireless ad-hoc network. In this protocol, the routing table is stored at each node containing the information required to packet transfer from one node to another in a network. This routing table has the information about the addresses of each of the other nodes in a network, the address of the next hop to reach the node, the routing matric, and the route sequence number, which is generated by the destination node. As the network topology changes, each node updates its routing table and broadcasts the routing table update packet to other nodes in the network [38].

In this routing protocol, each node has to store and update the routing table and requires an extensive network bandwidth for the update procedure. Therefore, these routing protocols are not useful for the UAV network.

Ad-hoc On-Demand Distance Vector (AODV) Routing: AODV is a reactive routing protocol for the wireless ad-hoc network and uses the bidirectional link for communication. In this protocol, if a communication link is broken between nodes, it does not affect the packet transmission between the source and the destination node, and no global broadcasting occurs. This routing protocol uses the route discovery cycle for route finding. In this protocol, the sequence number is used for loop prevention, and it also provides unique multicast communication. The routing table used in this protocol contains the information regarding the destination address, the next hop address, the destination sequence number and the lifetime. Each node in a network maintains a list of predecessor nodes to route through them. The lifetime in the routing table is updated every time the route is used in this protocol, and if the route is not used within its lifetime, then it expires [39].

The use of AODV routing protocol in the UAV network introduces a delay for route construction and route discovery in the network, and it also requires more bandwidth as the number of UAV nodes in the network are very high.

Temporally Ordered Routing Algorithm (TORA): This is an on-demand topology-based routing protocol that is used for communication in the ad-hoc wireless network. This protocol maintains a direct cycle graph for the path selection from one node to another in the network. In this protocol, a source node starts the communication by sending the route request packet to the neighbouring node. The neighbouring node then checks the packet header, and if the packet does not belong to it, it then rebroadcasts the packet according to the direct acyclic graph details. This process is followed until the packet is received by the destination node [40].

Dynamic Source Routing (DSR): The DSR protocol is known as a reactive or on-demand protocol and has been designed mainly for the MANET. In this routing protocol, the route discovery cycle is used for route finding. The entire route is stored in the packet header, and each node uses caches to store the route. The route discovery process is used in this protocol to find the path between the source and a destination node in the network. The source node then floods the route request (RREQ) with a sender address, destination address and Req-ID. In route discovery, each node appends its identifier when forwarding the RREQ [41].

In DSR, each node carries addresses of all the nodes from source to destination. Therefore, it is hard to implement in the UAV network, where the number of UAVs is very high.

Dynamic MANET On-demand (DYMO): DYMO is a reactive routing protocol based on the ADOV routing protocol. DYMO stores information of all the intermediate nodes from a source to a destination as a part of the whole route. It works efficiently in highly dynamic scenarios by monitoring communication routes. This routing protocol implements three messages: RREQ, Route Reply (RREP) and Request Error (RERR). A RREQ message is used by the source node to discover the valid route to reach the destination node. The RREP message is used to give the reply from the destination to the source node in the network. The third request message is used in this routing protocol to indicate the invalid route. This routing protocol performs two operations: route discovery and maintenance for communication between nodes in the network [42].

<u>Position-Based Routing Protocols</u>

The position-based routing protocols use a Global Positioning System (GPS) information to forward a packet from one node to another in the network. In these protocols, the source node initiates the packet forwarding to designation node and the position of the destination node is already known to the source node. This routing protocol is divided into three categories: Non Delay Tolerant Network (NDTN); Delay Tolerant Network (DTN) and the hybrid, which is a mixer of DTN and NDTN. The non-DTN is further classified as beacon base, non-beacon base and hybrid. The beacon base non-DTN routing protocol is furthered divided into the overlay and non-overlay routing protocols, as shown in Fig. 2.7.

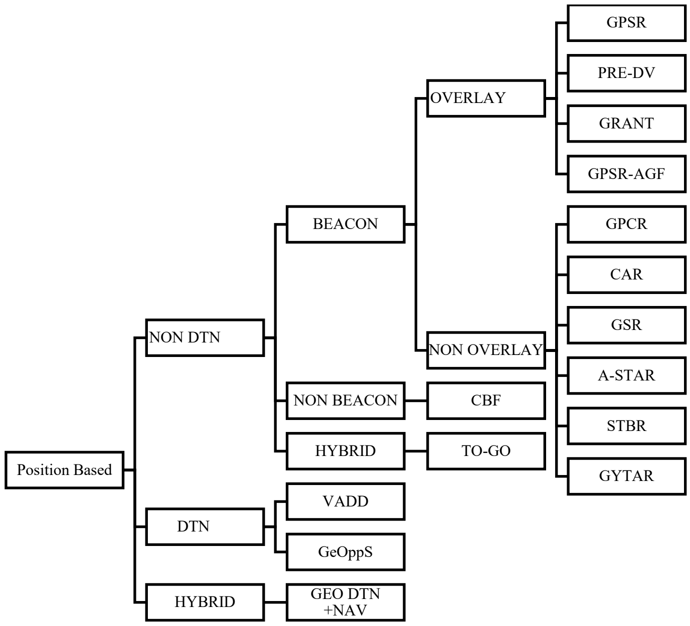


Fig. 2.7 Position-based routing protocols

Vehicle-Assisted Data Delivery (VADD): VADD is a DTN, position-based routing protocol. In this protocol, a moving node sends a data packet to a static node only if it has a guaranteed neighbour node. This protocol has three packet modes (that is, straightway, intersection and destination). The straightway mode sends the data packet to the next intersection, the packet forwarding is performed in intersection mode via a selection of the optimal path, and the packet is broadcast to the destination node in destination mode. This protocol is suitable for multi-hop data delivery; however, it faces a large delay in a big network due to rapid topology change [43].

Geographical Opportunistic (GeOpps) Routing: Geographical opportunistic (GeOpps) routing is a DTN-type position-based routing protocol. The operation of this protocol is based on a navigation system. During the communication, a source node follows the navigation system to select the next nearest node to the destination and forwards the data packets to that node. In this protocol, a node stores its data packets until the route information to the next node is no longer available. This protocol has a high packet delivery ratio. However, privacy is a big concern in this protocol due to the information of the navigation system, which is already known to each node in the network [44].

Greedy Perimeter Stateless Routing (GPSR): GPSR is a non-DTN, non-overlay position-based routing protocol used for wireless communication. It uses a greedy packet forwarding mechanism to send a data packet from the one node to another. Beacon message is used in his protocol to select the nearest node. The data packet forwards to the destination node using the greedy forwarding method, that is used to select the next node. If this method fails, this protocol selects the perimeter forwarding method to choose the next node. The recovery mode of GPSR is active only when the data packet is reached at a local minimum. In the recovery mode, the data packet is forwarded to the node, which is nearest to the destination node and, as such, the data packet reaches the local maximum. In this protocol, the sending node keeps the information of the next hop node only and, therefore, the packet forwarding is much easier. However, if the network is very large, network maintenance is a big issue due to the neighbour table becoming outdated [45].

GPSR+ Advanced Greedy Forwarding (AGF): The GPSR AGF protocol is an advanced version of the GPSR protocol and, as such, has improved the shortcoming of the GPSR protocol. It is a combination of GPSR and the advanced greedy forwarding method. The data packets of this protocol consist of speed, the direction of the node, overall travel time and processing time. In this routing protocols, during the communication between the source and a destination node, the intermediate node is updated with regard to the information from the destination node. The information of the unreachable node is also easily detectable in this protocol [46].

Position-Based Routing with Distance Vector (PBR-DV): A distance-vector algorithm is used in PBR-DV position-based routing protocol for communication between nodes in the ad-hoc wireless network. In this protocol, when the data packet reaches a local maximum, it uses the AODV protocol route discovery mechanism. Whenever a node receives the data packet, it checks whether the data is in range of the local maximum or nearest to the destination node. If this is not a case, the receiver node stores the packet information of the sender node. The receiver node either broadcasts the packet again or returns it to the node it came from [47].

Greedy Routing with Abstract Neighbour Table (GRANT): The greedy routing is used in this protocol by maintaining an abstract neighbour table for communication in the ad-hoc wireless network. Every node in the network contains information about its neighbourhood hops. The role of the abstract neighbour table is to divide the complete plane into different areas, and each area contains one representative. This protocol uses metrics to select the next forwarding node. This matrix depends on multi-hop neighbours and the multiplication of distance between nodes. The main advantage of this protocol is route recovery, where a packet takes less time to recover the route in this protocol [48].

Greedy Perimeter Coordinator Routing (GPCR): GPCR is an overlay network position-based routing protocol. This protocol is used for inter-vehicle communication for urban environments in which vehicle density is extremely high. Furthermore, it uses the position parameter to send the data packet to a target node. When a node wants to send their data packets to the target node, it selects the next node nearest to the target node using the position of that node. Each node in the network knows the details of its position, the details of their next neighbour, as well as details of the target node. This protocol works on two strategies known as greedy packet forwarding and repair strategy. In this protocol, information regarding the junctions and streets is collected from a planar graph rather than a street map and the routing is performed at the junction. The data packet forwards to the destination node using a greedy packet forwarding strategy and the repair strategy is introduced when the link is broken [49].

Connectivity Aware Routing (CAR): CAR is used for vehicle to vehicle communication in the highway-vehicle scenario. This protocol works in four modes: route discovery, forwarding packets, error recovery and path maintenance. In this protocol, guards are used for finding the current location of a target vehicle, and this binds the location with the help of geographical information. Temporary vehicle details are observed by the standing guards in this protocol [50].

Geographic Source Routing (GSR): GSR is a protocol based on route maps. In this protocol, the junctions and routes are converted into a graph using the route map. In this graph, the junctions represent the vertices, and the routes represent the edges. This graph is used to locate the shortest path from a source to a destination node. The data packet is transmitted between two junctions as part of communication through this routing protocol. If there is no connection available between nodes, this protocol uses the greedy forwarding and recovery mode strategy to transfer the data packet between nodes. The packet delivery ratio in this protocol is strong compared with other position-based routing protocols; however, this protocol does not work well within a sparse network [51].

Anchor-Based Street and Traffic-Aware Routing (A-STAR): A-STAR is a position-based routing protocol which works on overlay networks. This protocol is used for inter-vehicle communication within city networks. In this protocol, the anchor is identified with the help of a street map and the anchor route is then calculated using traffic awareness. This protocol uses two maps (that is, static and dynamic) to discover the route in the route discovery mode. The static map converts the city map into a graph and checks for a stable route; whereas, the dynamic map tracks information pertaining to real traffic. In this protocol, the data packet is transferred through the high-connectivity route [52].

Street Topology Based Routing (STBR): STBR is based on street topology. It works in three states. The first state is known as the master node, which is selected on a junction; the second is the slave node, which is another communication node on another junction; the third is the forwarding node, which is the intermediate node on the junction and lies between the above two junctions—in this protocol, the master node is chosen to check the link available to the next junction. The master node also works on two levels via a neighbour node to its direct junction node, as well as via a neighbour node to its junction node [53].

Greedy Traffic-Aware Routing (GyTAR): To overcome the local maximum problem, this protocol uses the carry and forward technique. A data packet is forwarded to junctions with the help of a greedy routing strategy. This protocol uses a digital map to send the data packet from a source to a destination node. Each junction is allocated a score based on the destination distance and density of traffic, and the data packet is then forwarded to the next junction using this score. The junction that receives the maximum score is the junction where the packet is to be forwarded [54].

Contention Based Forwarding (CBF) Routing: CBF is a non-beacon-based routing protocol. In this protocol, if a packet is ready to transfer, it finds its neighbour using geographical routing. The data packet is then forwarded to a neighbour node which is directly connected. This neighbour node decides the packet forwarding with the help of packet information. Each packet contains information regarding the position of the node where the packet comes from, the node ID, the destination, and the packet ID. This protocol saves a lot of bandwidth, as a beacon message is not used for packet transmission [55].

Topology-Assist Geo-Opportunistic (To-Go) Routing: To-Go routing is a hybrid approach protocol which combines beacon and non-beacon-based routing protocols. It is based on geographical routing, in which the target node is identified with the help of topology knowledge. Each data packet sending node possesses this topology knowledge. This protocol performs well in a high-density network. Furthermore, it uses the opportunistic forwarding technique to transfer the packet. Therefore, the packet delivery ratio is always higher in this protocol. Finally, all nodes are connected to each other and, therefore, there is no hidden terminal problem occurs in this protocol [56].

Geographical DTN with Navigation (GEO DTN + NAV): Geographical DTN with navigation (GEO DTN + NAV) is also a hybrid protocol, combining DTN and non-DTN position-based routing protocols. This protocol works in two modes: DTN and non-DTN. The change in mode depends on the connectivity of the nodes and other issues, such as how many nodes are traversed by a data packet, the node direction, and the destination node. This protocol uses a virtual network interface (VNF) to provide information regarding the forwarding node and the route node [57].

2.5.6.3. The adaptability of existing ad-hoc routing protocols in UAV network

In the previous section, various wireless ad-hoc routing protocols are outlined. Some of these routing protocols were designed especially for the MANET and VANET. Each of this routing protocol has its own unique capabilities and characteristics. Keeping all the challenges associated with the UAV network in mind, we analysed their adaptability and suitability for the UAV network. Table 2.3 presents a comparison between these routing protocols based on their adaptability in the UAV network.

Table 2.3 Ad-hoc routing protocols adaptability with the UAV network

Protocol	**Type**	**Problems for adaptability with UAV networks**
LCAD	Static	Delay in delivery
Data Centric	Static	Network overload
MLHR	Static	Single point of failure
DSDV	Proactive	High overhead and high bandwidth consumption
OLSR, GSR, FSR	Proactive	Routing loop problem and generates high overhead
DSR	Reactive	Scaling and dynamic network is a problem due to the maintenance of a complete route address

AODV	Reactive	Produces delay in organising a route and further delay in link failure
ZRP	Hybrid	High complexity and difficult to maintain radius in UAV network
TORA	Hybrid	Invalid result produced
VADD	Position-based	A significant delay in high-density traffic
GeOpps	Position-based	Privacy issues due to navigation detail disclosure
GPSR	Position-based	Route maintenance is hard in high-route length path
GPSR + AGF	Position-based	Does not provide an optimal solution for route selection
PRB-DV	Position-based	An additional overflow of packets
GRANT	Position-based	Flooding range is small
Car	Position-based	Cannot adjust different subpaths
GSR	Position-based	Not suitable for sparse network
A-STAR	Position-based	Low packet delivery ratio
STBR	Position-based	High complexity
TO-GO	Position-based	End-to-end latency is higher
GeoDTN + Nav	Position-based	Difficult to select the next node in the network

As we can see from the about Table 2.3, each of the wireless ad-hoc routing protocols that were designed for the MANET or VANET have some problem to use them in the UAV network for communication between UAVs. There are also some other VANET communication protocols proposed in the past research and we studied them to understand their adaptability for the UAV network. However, some of these existing wireless ad-hoc routing protocols were used in FANET for UAV to UAV communication with some modification according to the network scenario. As such, the communication protocols for VANETs and UAV networks proposed in the past research are discussed in the next section.

2.5.7 Communication protocols for VANETs and UAV networks

In Section 2.5.6, we discussed the wireless ad-hoc routing protocols and their adaptability for the UAV networks. Apart from those communication protocols, some other communication protocols were also proposed and implemented for VANETs and UAV networks in past research. In this section, we review such of those protocols that were designed for the special scenario of VANET and UAV network.

2.5.7.1 VANETs communication protocols

As mentioned earlier, the communication between UAVs in the UAV network is similar to the vehicle to vehicle communication in VANET. As, in both of these networks, the moving nodes (that is, UAVs or vehicles) communicate with each other with the help of a wireless medium. The main challenge associated with the VANET is the communication protocols that are used to communicate between vehicles and vehicle-to-RSU. Various VANET communication protocols have been proposed and implemented for V2V and V2I communication, where some have been tested on real vehicles, while others have been evaluated using network simulation.

Dikaiakos, et al. [58] proposed location-aware services that can help to inform the car driver about traffic condition and roadside facilities. Vehicular Information Transfer Protocol (VITP) [58] is a location-aware application layer stateless communication protocol which is based on the client-server model. VITP defines the format for query and replies messages that exchange between vehicular clients and servers. It is used to define syntax and semantics of messages which handle the queries and replies between nodes in VANET.

The Diffie-Hellman protocol [59] is a secure communication protocol and a novel approach in car-to-X (C2X) communication. The Car-to-Car and Car-to-Infrastructure communication system form a C2X network. This protocol is considered as a secure protocol assuming that the discrete logarithm problem is intractable. In the implementation of this protocol, a group of vehicles is defined first by polling the neighbourhood table. Thereafter, key fragmentation is defined for each vehicle by selecting the secret number. A group key is then calculated and shared between the communicating vehicles. Every group member uses this group key to sign the messages.

A communication system is described in [60] which integrates the two systems, V2V and V2I. In this system, V2I is concerned about IPv6 network mobility, and V2V uses the hybrid solution that is based on intelligent delivery and delay-tolerant network. IPv6 has the large address space required by VANET for supporting the millions of entities which can easily exchange information during the communication. The V2V communication protocol uses the 802.11p interface to transmit packets between nodes.

Yang, et al. [61] proposed a vehicle collision warning communication (VCWC) protocol which addresses the problem of achieving low latency in delivering emergency warning messages (EWMs) for various road situations. The physical characteristics of the channel follow the 802.11b standard, with a channel bit rate of 11Mbps. Whenever an abnormal vehicle supplies an EWM, the out-of-band busy signal is raised that is sensed by the vehicles within a distance of two hops. This rate-decreasing algorithm for EWMs helps to achieve the real-time transmission of EWMs.

The above mentioned VANET communication protocols cannot be directly used for communication in the UAV network. This is because, in VANET, nodes (vehicles) can contain high-speed processors, and this feature is not currently available in UAVs. Adding a high-speed processor to a UAV poses several challenges. First, it will reduce the flight time of the UAV, as it draws additional power from the batteries and weight of the processor, and the accessories used will also reduce the maximum payload of the UAV. Additionally, UAVs need to route real-time videos through the UAV network to the ground station, requiring them to be engaged in more data transfer than the nodes in VANET. As such, it seems appropriate, and is probably a requirement, to develop a standardised inter-UAV communication mechanism for the UAV network.

2.5.7.2 Communication protocols for UAV networks

Some researchers have already made efforts towards developing communication protocols in the area of UAV communication. A UAV search mission protocol (USMP) was proposed by Lidowski, et al. [62], which employed a combination of inter-UAV communication and geographic routing. In this study, location update and waypoint conflict resolution served as key areas. The location update feature of USMP works with two design methodologies. In the first design, the message is explicitly passed to neighbour UAVs, and in the second design, the location of GPSR information is reused. The waypoint selection is performed using the reservation message in this protocol. Furthermore, USMP implementation was based on key parameters such as transmission power, swarm size, sensor type and initial location. The results of this study were generated using the analysis of variance (ANOVA) and the outcome changed the hypothesis which stated that the search performance of UAVs would be improved by geographic routing for waypoint conflict resolution. The results of this study also demonstrated that GPSR harvesting is not a replacement for an explicit location update.

Tuna, et al. [20] proposed a communication system for post-disaster management. In this system, the UAV contains an onboard computer containing three subsystems for end-to-end communication, formation control and autonomous navigation. Through the use of protocol simulation, the findings of this study demonstrated that this communication system would be a feasible solution for disaster recovery and helpful in disaster management.

Alshbatat and Dong [63] proposed the Adaptive Medium Access Control Protocol for UAV communication (AMUAV). This protocol is based on the ad-hoc network that used a directional antenna for transmission between UAVs. For this approach, each UAV has a primary directional antenna mounted on its top and a secondary directional antenna fixed underneath it. There are also omnidirectional antennas as in other UAVs. Both omnidirectional and directional antennas participate in the communication between two UAVs. The switching of transmission from omnidirectional to directional antennas was undertaken with the help of an information table. This protocol solved the problem of communication of UAVs among themselves without the need for a ground station. A GPS and inertial measurement unit (IMU) were used in this protocol to update the location of a UAV. AMUAV packet transmission was achieved using the IEEE 802.11 standard, and an information table for message passing was updated using a direct network allocation vector (DNAV). With this approach, the AMUAV protocol initiates the data packet transfer and checks the distance between two UAVs. If this distance is less than the range of the omnidirectional antenna, the data packet is then sent by this antenna. Otherwise, the MAC layer checks the altitude of the first UAV and compares it with that of the second UAV. If this is less than or equal to the altitude of the second UAV, the data packet is then sent by the primary directional antenna, and the direction will be towards the second UAV. On the other hand, if the altitude of the first UAV is greater than the altitude of the second, the data packet is then sent by the secondary directional antenna and the direction will be towards the second UAV. After successfully receiving the data, the second UAV sends the acknowledgement to the first UAV using the omnidirectional antenna. These study findings demonstrate improvement in performance through switching between the two antennas compared with the use of only omnidirectional antennas [63].

Martins, et al. [64] proposed an inter-module communication (IMC) protocol for communication between heterogeneous vehicles, sensors and human operators. They designed the IMC protocol by testing the various vehicles, such as Autonomous Underwater Vehicles (AUVs), Autonomous Surface Vehicles (ASVs), UAVs, and Remotely Operate Vehicles (ROVs). This IMC protocol exchanges the real-time information between these vehicles and the human operator.

The above-mentioned protocols for the UAV network as well as the exiting wireless ad-hoc routing protocols are not suitable for the communication in the private UAV network where all UAVs belong to the one person or organisation. As the number of UAVs are comparatively low in the private UAV network, the finding the next neighbour node through routing is always be a challenging task. Furthermore, to implement routing in the private UAV network, many UAVs need to be deployed in the same geographical area and it would increase the cost of the network. As such, it is not an appropriate solution for communication in the private UAV network. Therefore, it is required to investigate and a better communication approach in the private UAV network need to be implemented.

In a private UAV network, communication between UAVs and UAV to the ground station plays an important role. In this network, control and data signals are transferred from one UAV to another when attempting to reach its destination. A UAV in this network can send or receive these signals to other UAVs or the ground station through the intermediate UAVs. As mentioned earlier, the communication protocols discussed in Section 2.5.6 and 2.5.7 are not suitable to use for UAV to UAV and UAV to the ground station communication in this private UAV network, as it has comparatively low UAV nodes and all of them belonging to the same owner. Therefore, a suitable communication mechanism between UAVs and UAV to the ground station in this network should be implemented by keeping the specifications of this private UAV network in mind for fast control and data transmission. In this UAV network, the transmission of these signals between the UAVs and the UAV to the ground station will be carried out using Wi-Fi channels on the 2.4 GHz or 5 GHz frequency spectrum. The currently available micro UAVs in the market have been designed for one-to-one communication only (that is, UAV-ground station); however, we used these available UAV models in this research to test the communication between UAVs and UAV to the ground station by forming two private UAV networks and, each of the UAVs in this network will also use communication channels of these frequency spectrums to perform UAV-to-UAV and UAV to the ground station communication. As such, it is required to investigate the capability of these Wi-Fi communication channels to use in this network.

Signaling mechanisms for communication between UAVs and UAV to the ground station in the private UAV network is to introduce another area of work in this research. Accordingly, observing how this currently available signaling mechanism may work for this UAV network is important. Thus, change in the signaling mechanism to make it more efficient to use in a private UAV network should be investigated. We investigated all of these issues and proposed the solutions for them based on the requirement of the private UAV network in this research.

To test the communication between UAVs and UAV to the ground station, it is required to implement a private UAV network. A private UAV network can be formed by connecting the number of micro UAVs in the network. In the next section, we discuss the features and capability of two UAV models that were used in this research to test the communication in the private UAV testbed networks.

2.6 Micro UAVs for private UAV network

To form a private UAV network, several networking components are required, and the main component is the micro UAV model. At present, two major companies DJI and Parrot, are designing and manufacturing these UAV models. These companies introduce different types of UAV models annually by adding enhanced functionality to each UAV model. These two companies aside, other companies exist in the market and also contribute to the UAV industry by developing UAVs on an annual basis. With the aim of the current study to implement a better communication UAVs and UAV to the ground station by forming the private UAV network, we investigated the functionality and capability of many UAV models. The details of several UAVs, such as Parrot, DJI, 3DR, Storm and other UAVs from major companies, are outlined in the Appendices of this thesis [8], [10], [11]. Following the investigation of these various UAVs, we settled on two models for the implementation of two private UAV networks for our study: Storm 4 Mini UAV was selected according to the research budget allocation, and it can carry the sufficient payload, and Parrot AR 2.0 UAV was bought for some other project and was available to use later in this research. The following subsection provides details with each of these UAVs separately [9], [10].

2.6.1 Storm 4 Mini UAVs

For this study, we searched the UAVs to form a private UAV testbed network which fit within our research budget. The biggest issue in this process concerned with the UAV payload. Accordingly, we were interested only in those UAVs which were capable of lifting a certain weight. Thus, we decided the Storm 4 Mini UAV would be suitable in forming the private UAV network for our study. This UAV is a remote controller-based model with the dimensions of 490mm x 490mm x 140 mm. Fig. 2.8 illustrates the Storm 4 Mini UAV with its controller. The take-off weight of the Storm 4 Mini UAV is 850 gm, and it can carry a maximum 250 gm payload. This UAV uses 11.1 V Lithium polymer batteries, which allow them to fly for 8 to 12 minutes. The actual flight time of this UAV depends on the payload carried during the flight, as well as the environmental conditions at the time of flying. For example, if there are windy conditions and the UAV is carrying a maximum payload, a flight time of only 4 to 5 minutes is likely. This UAV model is controlled by a 2.4 GHz Devo controller. Every Storm 4 Mini UAV model is bound with its remote controller, and a UAV operator can fly this UAV with knowledge of control function of its remote controller. As the UAV industry is growing rapidly, these UAVs are now obsolete from the market. Accordingly, Storm company is manufacturing upgraded models of this UAV, and the old model (that is, the Storm 4 Mini) is no longer available in the market [9].

Fig. 2.8 Storm 4 Mini UAV, Source: Storm 4 Mini UAV, http://www.helipal.com from [9]

2.6.2 AR 2.0 UAVs

The AR 2.0 UAV is a product of a company named Parrot. These UAVs were bought for some other project but were available later to be used in this research. As mentioned previously, the main objective in the selection of a UAV model was the payload. However, another reason associated with the selection of the AR 2.0 UAVs for this study as they are smartphone-controlled UAVs and both control and data signals in this UAV are transmitted through the Wi-Fi frames. The AR 2.0 UAV uses Wi-Fi communication to connect the UAV to the ground station. This UAV model can be controlled using a smartphone, iPad or laptop which contains a Wi-Fi interface. Parrot introduced the AR 2.0 UAV with a flight-controlling application, and this application can be freely downloaded from the Apple store or Google Play store. Once we plugged the battery into the AR 2.0 UAV, the Wi-Fi Service Set Identifier (SSID) appeared on our smartphones. First, we need to connect our smartphone with the AR 2.0 Wi-Fi network SSID, and then we can fly this UAV with the installed application. The AR 2.0 UAV model is shown in Fig. 2.9.

The AR 2.0 UAV model uses a Wi-Fi control packet to control the UAV from the ground station, as well as to send the video and feedback packet to the ground station. AR 2.0 UAV contains two cameras and a live video stream captured by these cameras can be watched on the smartphone application. The flight time of this UAV is 12 minutes, but the actual flight time depends on the payload and weather conditions at the time of flight. This UAV model is capable of carrying some payload, and we determined in our study that it can easily carry 75-100 grams. The High Definition (HD) quality live video stream is also advantageous to use this UAV to form a private UAV network as it can send a good quality video to the ground station. Another reason for the selection of this UAV was its open-source software development kit to develop an application for this UAV. Therefore, modifications to controlling the UAVs from the ground station could be performed to test the communication from one UAV to another based on the requirements of this study. These UAVs are still available in the market; however, Parrot has ceased introducing and manufacturing upgrades to this particular UAV model [10].

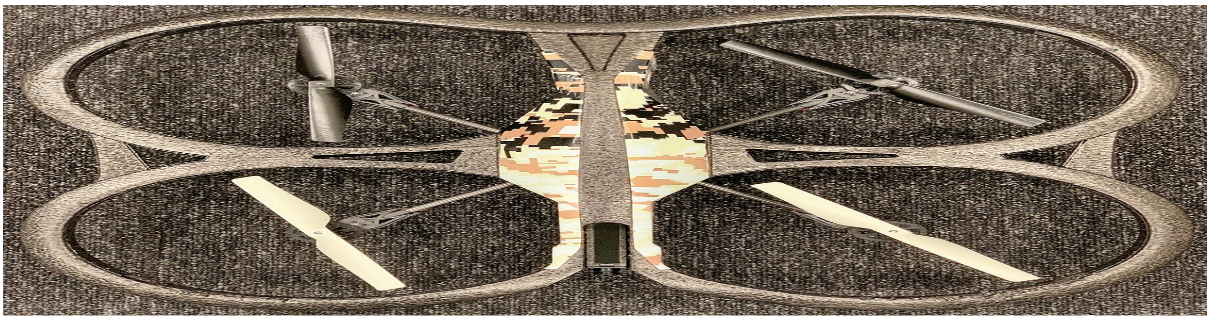


Fig. 2.9 AR 2.0 UAV model

At present, most of the new UAV models that are operated through smartphones or tablets use IEEE 802.11 standard to connect the UAV with the ground station, which works over the 2.4 GHz or 5 GHz radio frequency band. The camera mounted on these UAV transmits videos to the ground station using a wireless signal which also operates on a 5 or 2.4 GHz frequency [8], [10], [11]. The main issue associated with these UAVs is that they cannot send their control and data signal to other UAVs and they are bound to work with their respective ground stations only. Research undertaken by Yue [65] demonstrated the communication between a UAV and ground station uses two separate wireless links. The wireless link used to control the UAV operates on a 2.4 GHz frequency; whereas, the wireless link used for transmitting data from a UAV to a ground station (for example, video streaming) works via a 915 MHz frequency [65]. DJI commercial UAVs also use two wireless links (that is, 2.4 GHz and 5 GHz) to transmit control and videos from a UAV to a ground station [8]. Parrot AR UAV is another example which uses these wireless frequencies to transmit their control and data signals [10]. Therefore, the private UAV network for the commercial UAV applications also uses these frequency spectrums to transmit the control and data signals between UAVs. It is a question to investigate which frequency spectrum should be used to transmit these signals between UAVs in this private UAV network and how these UAVs can be used to form a private UAV network as they cannot send the control and data signals to another UAV. All of these issues are considered in this research and private UAV testbed networks were implemented to test the communication between UAVs and UAV to the ground station that can be used in the commercial UAV applications.

2.7. Chapter summary

The aim of this research is to implement better communication between UAVs and UAV to the ground station in the private UAV network. For this purpose, it is required to get complete detailed information about the UAVs, their applications, the existing UAV network and communication protocols. This chapter outlined the background and significance of communication in the existing UAV networks. This chapter began with an introduction of UAVs and their types. We also covered the various commercial applications of UAVs and the requirement for the UAV network for these applications is also discussed in this chapter. Following this, MANET and VANET (sharing some similar characteristics with FANET) and existing UAV networks were discussed. We conducted a literature review and outlined the existing UAV communication protocols used in UAV networks. In addition, detailed information on the Storm 4 Mini and AR 2.0 UAVs were also discussed in this chapter, both of which were selected to implement the private UAV testbed networks to test the communication between UAVs and UAV to the ground station.

This chapter provides complete background information on the existing UAV networks and the communication protocols used in these networks. However, a private UAV network was found as a suitable network for commercial applications of UAVs where all the UAVs belong to one person. The main concern in this private UAV network is the implementation of a better communication system. To test this communication system, two practical testbed networks were implemented in this research. In the next chapter, we will discuss the detailed information about the implementation of the communication mechanism in a private UAV network.

Chapter 3

A testbed for testing communication in Private UAV networks

3.1 Introduction

In Chapter 2, we discussed the use of micro UAVs in the various applications domain. These UAVs can give better performance if they work together and form a UAV network. The existing UAV networks and the communication protocols used in them are also discussed in Chapter 2. As mentioned in Chapter 2, FANET and IoDs networks are not suitable to use in the commercial application of UAVs where all UAVs belong to the same person or one organisation, creating a private UAV network. A private UAV network is different from the FANET or IoDs as it has a relatively fewer number of UAVs and they belong to the same owner.

The main purpose of this chapter is to discuss the communication in the private UAV network that was implemented in this research by forming two small private UAV testbed networks. These testbed networks were used to test the communication between UAVs and UAV to the ground station. First, we tested the communication in the private UAV network by simulation software. For this purpose, we used computer simulations software and investigated how the messages could be passed efficiently between the UAVs in this network. In a private UAV network, a UAV communicates with other UAVs either directly or through the intermediate UAVs. As such, the routing or switching two approaches can be used for communication between UAVs and UAV to the ground station in this network. Since the number of UAVs in the private UAV network are comparatively low, these routing protocols are not suitable for communication between UAVs and UAV to the ground station. Therefore, in the private UAV network messages can be passed between UAVs or UAV to ground station by switching through the UAV nodes. As such, we implemented switching for communication from one UAV to other and the ground station in these private UAV testbed networks.

In this chapter, we discuss the implementation of two private UAV testbed networks that were used to test the communication between UAVs and UAV to the ground station. We faced lots of challenges during the implementation of communication between UAVs and UAV to the ground station over these practical testbed networks: both of the UAV models (Storm 4 Mini and AR 2.0) were not able to communicate with other UAVs, communication interference, adjustment of UAVs according to communication range, and AR 2.0 UAV payload capacity are few of them. In this chapter, we discuss each of these issues and the solutions that we adopted for them during the practical testbed networks implementation.

The remainder of the chapter is as follows. In Section 3.2, we discuss the communication mechanism for the private UAV network. We tested two private UAV networks with six UAVs to verify the communication between UAVs and UAV to the ground station through the simulator, and it is discussed in Section 3.3. The design of the two private UAV networks in the OPNET simulator is discussed in Section 3.4. Section 3.5 deals with the analysis of the simulation results for communication between UAVs and UAV to ground station in both private UAV networks. Different graphs and tables are provided to verify these communications through simulation in this section. In Section 3.6, we discuss many advantages to use switching over routing for these communications in the private UAV network. We outline the first private UAV network experimental testbed setup in Section 3.7. We test the communication between UAVs and UAV to the ground station in this network through various testcases, and they are discussed in Section 3.8. Section 3.9 describes the second private UAV network practical testbed setup, and the experimental results for test these communications in this UAV network are covered in Section 3.10. Finally, a chapter summary is given in Section 3.11 to conclude the chapter.

3.2 Implementation of the private UAV network to test the communication

The main purpose of this research is to implement better communication in a private UAV network for fast control and data signals transmission. For this purpose, we implemented the private UAV testbed networks and tested these communications in these networks. The first issue encountered at that time was to select the UAV models that could be used for the implementation of a private UAV network. According to our allocated research budget, we started a search for the UAVs that could be a part of this private UAV network design. The commercial micro UAVs that were available in the market could not communicate with other UAVs since they were specially designed for the communication between UAV and its ground controller. It had been cleared that we required some additional hardware devices with each UAV that could enable communication between UAVs. For this purpose, we decided to use an additional WAP with a power bank and a Wi-Fi camera with each UAV to implement this private UAV network. We moved our search direction to find commercial UAV models that can lift these additional devices payload and also fit into our research budget. We selected the Storm 4 Mini UAV and Edimax WAP to implement a private UAV network and bought only six of them due to the budget restriction. Accordingly, we defined a UAV node for this network is a node with a combination of a UAV, WAP, power bank and Wi-Fi camera. Fig. 3.1 presents a symbol of this UAV node finalised for this private UAV network; this UAV node has the capability to communicate with another UAV node in the network. Throughout this thesis, this UAV symbol is used to represent the UAV node for the private UAV network.

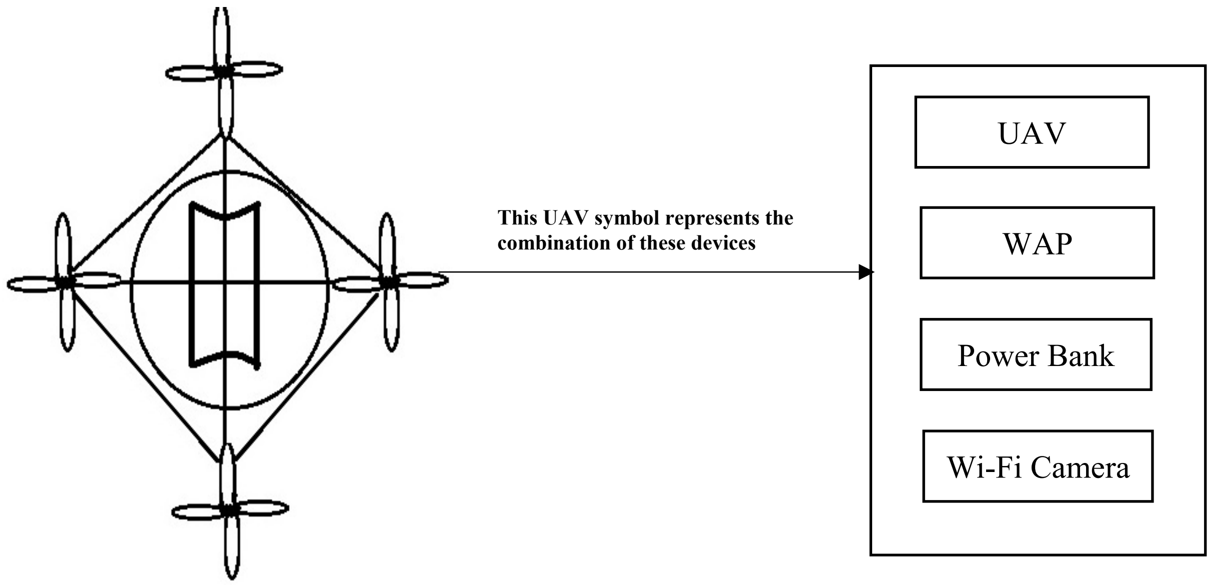


Fig. 3.1 Special UAV node with additional devices

While we were waiting for the UAVs and other hardware devices, we simulated this private UAV network to understand the network behaviour and verified the communication between UAVs and UAV to the ground station. The details about the network simulation are presented in the next section.

3.3 Simulation study of UAV communication in a private UAV network

Prior to implementation of any network, it is better to simulate the network first, to understand the network behaviour for the communication between devices. The Riverbed modeler academic edition OPNET simulator was used in this research to test the communication between UAVs and UAV to the ground station for the private UAV network. This simulator software is freely available to use for academic study and research. While the free academic version of OPNET has limited features, it still fulfilled our requirements to test the communication between UAVs for these UAV networks [66].

At the beginning of the research, it was not clear about the position of the UAVs in this private UAV network. So, we simulated this network with two different arrangement of UAVs to verify the communication between UAVs and UAV to the ground station. For the network simulation, we used six UAV nodes (similar number of UAV nodes for practical testbed network) and designed two private UAV networks in OPNET. In the next subsection, we discuss the positions of UAVs in the private UAV network that were used to form this network in OPNET simulator.

3.3.1 Position of UAVs in the private UAV networks

At the early stage of this research, it was difficult to fix the position of each UAV to form a private UAV network. It was a big question where to place each UAV within the network to get maximum utilisation from this network. Therefore, for the simulation study of this network, we designed two private UAV networks in the OPNET by placing the UAVs in a different position. In the first private UAV network simulation, we placed six UAVs three parallel to each other and formed a private UAV network. We connected UAVs in tandem for the second private UAV network simulation through OPNET. The two private UAV networks with their UAV positions are shown in Fig. 3.2 and Fig. 3.3. Aside from the network topology, both private UAV networks are similar.

Based on Fig. 3.2, in this first private UAV network, each UAV captures the video data of the ground while they are flying and sends it to the single ground station. Specifically, UAV 1, 2 and 3 can fly along the top edge and UAV 4, 5 and 6 can fly along the bottom edge so that their final positions will only be one hop to the right of their original positions. Since the position of each UAV and the communication between UAVs within this private UAV network, is fixed, each UAV can only communicate with its neighbour UAVs as shown by bidirectional arrows in Fig. 3.2. As such, UAV 4 can communicate with UAV 1 and UAV 5; UAV 5 can communicate with UAV 2, UAV 4 and UAV 6; UAV 6 can communicate with UAV 3 and UAV 5; the ground station can only communicate with the UAV 6.

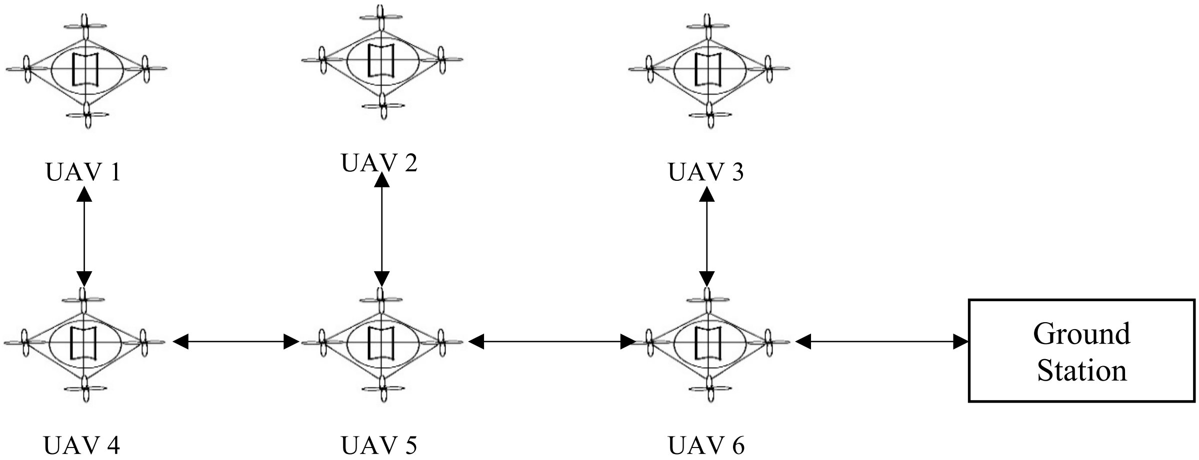


Fig. 3.2 A parallel arrangement of UAVs in a private UAV network

Fig. 3.3 presents the second private UAV network in which UAVs are connected in tandem. As shown in Fig. 3.3, each UAV can communicate with its neighbour UAVs. Therefore, UAV 1 can communicate with UAV 2; UAV 2 can communicate with UAV 1 and UAV 3; and so on. Furthermore, the ground station can only communicate with the UAV 6. In this private UAV network, the configurations shown in Fig 3.2 and 3.3 are only possible. Random movements of UAVs are not allowed in a private UAV network. Before the explanation of the simulation process, we need to discuss the communication path, which is followed by each UAV to transmit their video data at a ground station in both UAV networks.

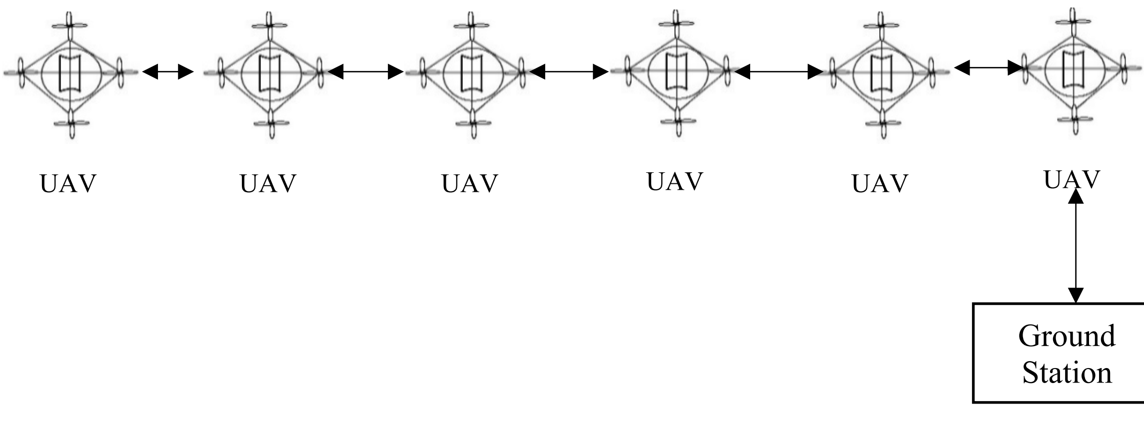


Fig. 3.3 Second private UAV network in which UAVs are connected in the tandem

3.3.2 Communication path for UAV to the ground station communication

In both private UAV networks, the position of each UAV was fixed, and they could only communicate with their neighbour nodes, as shown in Fig. 3.2 and Fig. 3.3 with bidirectional arrows. Therefore, the communication path for video data transmission from each UAV to the ground station can easily be defined for both private UAV networks and shown in Tables 3.1 and 3.2. Table 3.1 gives the communication path for each UAV to the ground station for video data transmission in the first private UAV network. As shown in the first row of Table 3.1, when UAV 1 sends video data to the ground station, this data transmits through UAV 4, 5, and 6 and reaches to the ground station. In this network, each UAV also added its own video data when it transmits the video data from the previous UAV to the next UAV.

Table 3.1 The communication path from UAV to the ground station in the first private UAV network

Source–> Destination	**Communication Path**
UAV 1→GS	UAV 1→UAV 4→UAV 5→UAV 6→GS
UAV 2→GS	UAV 2→UAV 5→UAV 6→GS
UAV 3→GS	UAV 3→ UAV 6→GS
UAV 4→GS	UAV 4→UAV 5→UAV 6→GS
UAV 5→GS	UAV 5→UAV 6→GS
UAV 6→GS	UAV 6→GS

Table 3.1 presents a very simple communication path for video data transmission between UAVs and the ground station in this network, as the number of UAVs is very few and each UAV location is pre-defined in the network. Additionally, Table 3.2 presents the communication path for video data transmission from each UAV to the ground station for the second private UAV network. As we can see from Table 3.2, the video data from UAV 1 passes through each UAV prior to reaching the ground station.

Table 3.2 The communication path from UAV to the ground station in the second private UAV network

Source–> Destination	**Communication Path**
UAV 1→GS	UAV 1→UAV 2→UAV 3→UAV 4→ UAV 5→UAV 6→GS
UAV 2→GS	UAV 2→UAV 3→UAV 4→ UAV 5→UAV 6→GS
UAV 3→GS	UAV 3→ UAV 4→ UAV 5→UAV 6→GS
UAV 4→GS	UAV 4→UAV 5→UAV 6→GS
UAV 5→GS	UAV 5→UAV 6→GS
UAV 6→GS	UAV 6→GS

3.4 Private UAV networks setup in OPNET

We discussed two private UAV networks in the last section with the position of each UAV in them. In this section, we explain the private UAV network setup in OPNET that was used during the simulation in order to verify the communication between UAVs and the UAV to the ground station.

The OPNET simulator does not provide any component that can be directly used as a UAV node to form this private UAV network. For the simulation of this network, we used a wireless Local Area Network (LAN) router to act as a UAV node as it can communicate with another wireless LAN router within the network. To generate the video traffic from each UAV, we added a wireless LAN workstation sensor device component with each UAV that acts as a UAV camera and can send the video traffic to its corresponding wireless LAN router. We also required a component for the ground station capable of receiving data from each UAV. Hence, we used the wireless server component as a ground station in order to implement the private UAV network in OPNET. Now we are going to discuss these components one by one and explain their role in the formation of a private UAV network in the OPNET simulator.

Fig. 3.4 represents wireless LAN workstation sensor device as the Wi-Fi camera of each UAV node used in the simulation. This component belongs to the wlan_wkstn_adv modeller family in OPNET, and it is a wireless workstation sensor device that fulfilled the requirement to generate the video traffic for each UAV node in this private UAV network setup. Each wireless client was given a node name (that is, from ‘sensor device 1’ to ‘sensor device 6’).

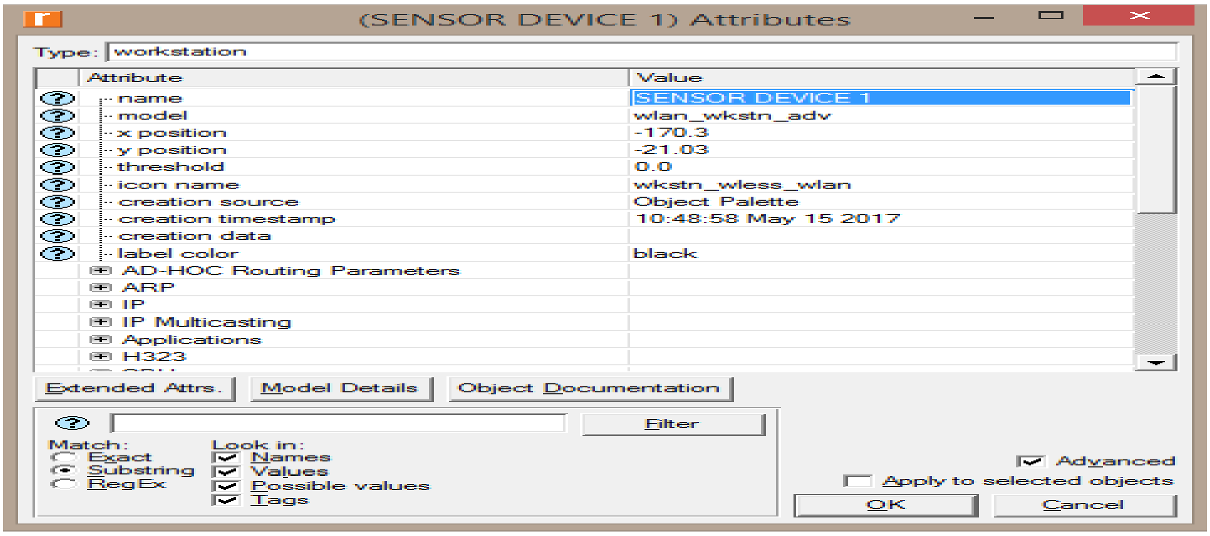


Fig. 3.4 UAV node in the OPNET

We required a WAP to be connected to each UAV that could communicate with the WAP of the other UAV in this private UAV network. For this purpose, we used the wlan2_router_adv model component in the OPNET. We gave the name of these components from ‘UAV 1’ to ‘UAV 6’. One of these components is shown in Fig. 3.5 (below).

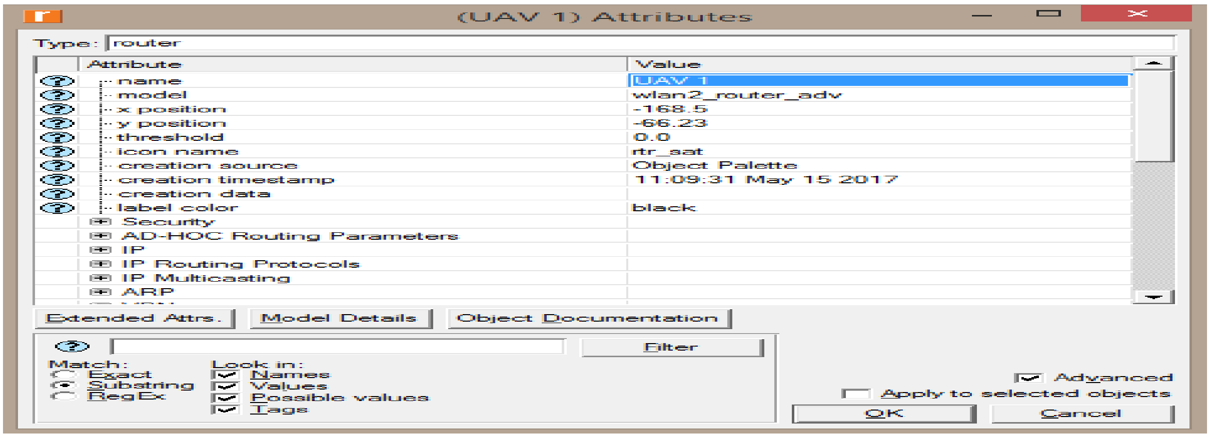


Fig. 3.5 UAV’s WAP in OPNET

The ground station of this private UAV network that was used in the OPNET is shown in Fig. 3.6. We used the wlan_server_adv modeller family of the OPNET to represent the ground station of this network. With all the required components in place, we were then able to begin setting up our first private UAV network design in the OPNET. Fig. 3.7 shows the complete first private UAV network design in the OPNET simulator.

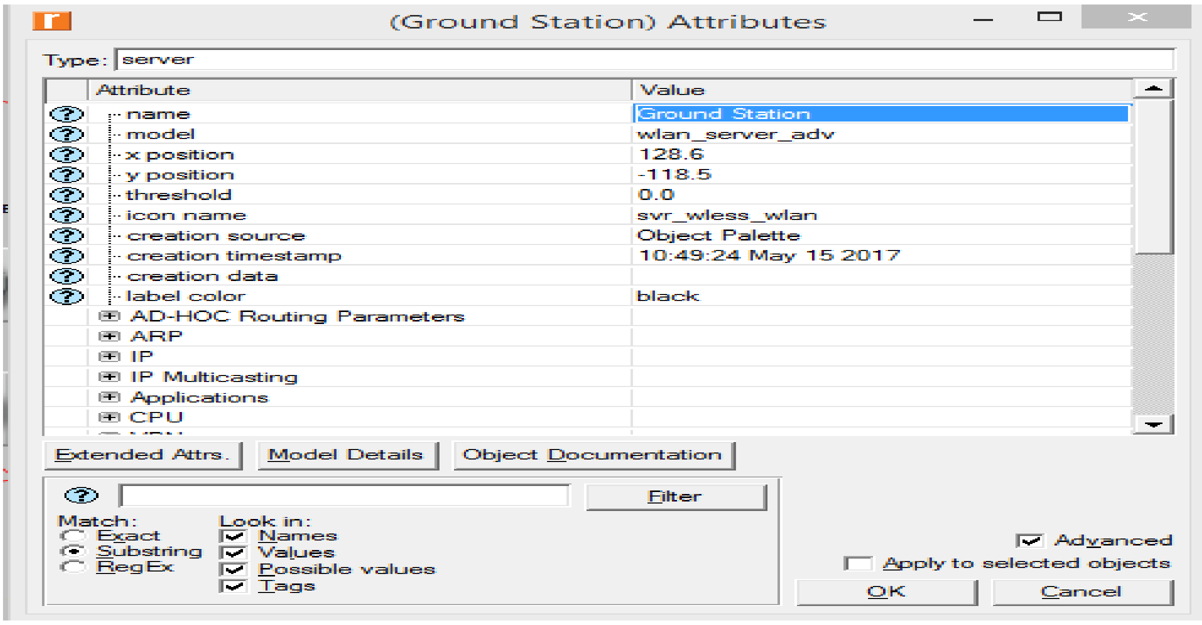


Fig. 3.6 A ground station in the OPNET

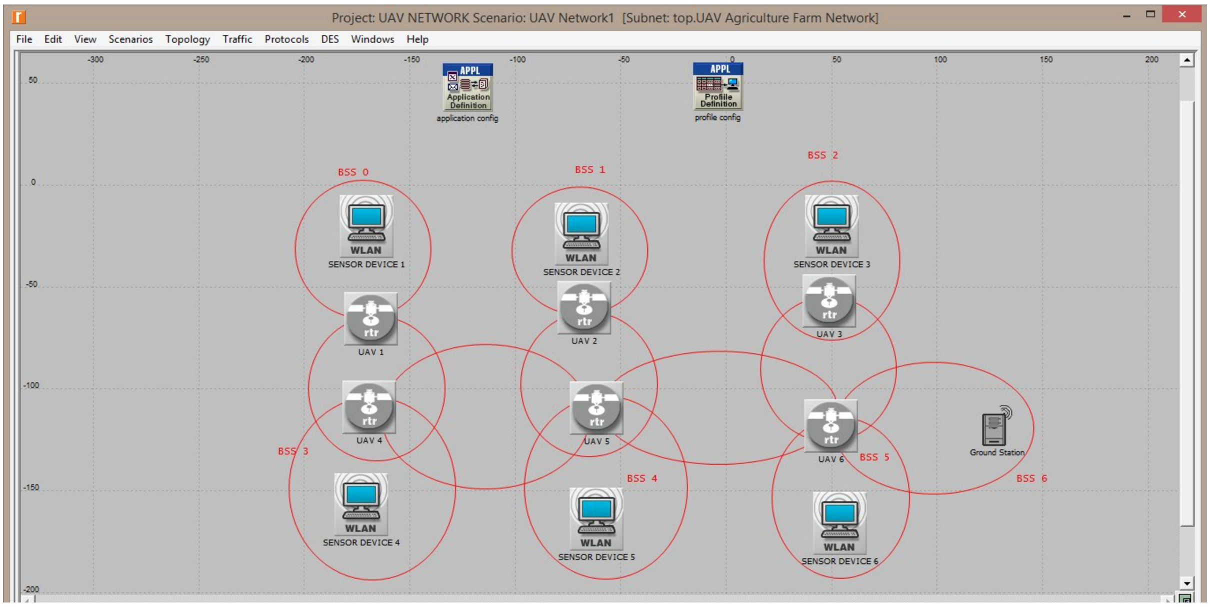


Fig. 3.7 Private UAV network with a parallel arrangement of UAV in OPNET.

From Fig. 3.7, we can see that three UAVs (UAV 1, 2 and 3) are arranged in the top row, and three UAVs (UAV 4, 5 and 6) are in the bottom row in this private UAV network. We configured each UAV node in the OPNET in such a way that each UAV had a path to the ground station through the neighbour UAVs, as shown in Fig. 3.7. Furthermore, in this private UAV network, UAV 4 receives data form UAV 1 and transmits it to UAV 5, then UAV 5 transmits this data to UAV 6, and the ground station receives data from the UAV 6. It should be noted that each intermediate UAV also adds its own data during this communication. Similarly, each UAV sends its data to the ground station in this private UAV network. For this private UAV network simulation, we also required two models (known as the application configuration and profile configuration) to generate the traffic at each UAV. We configured each UAV in this private UAV network to support a profile configuration which could have multiple applications. For this purpose, we first configured the application configuration model and defined the custom application to generate the video data. To achieve this, we added a new row in the application definition of this model for the video data. The configuration setup of this model is shown in Fig. 3.8. In this network simulation, we could configure each sensor device to support these applications. Rather than adding an application one by one to each sensor device, we used profile configuration for this purpose and added all the supported applications into a single profile and then configured each sensor device to support this profile. The configuration setup of the profile configuration model is given in Fig. 3.9.

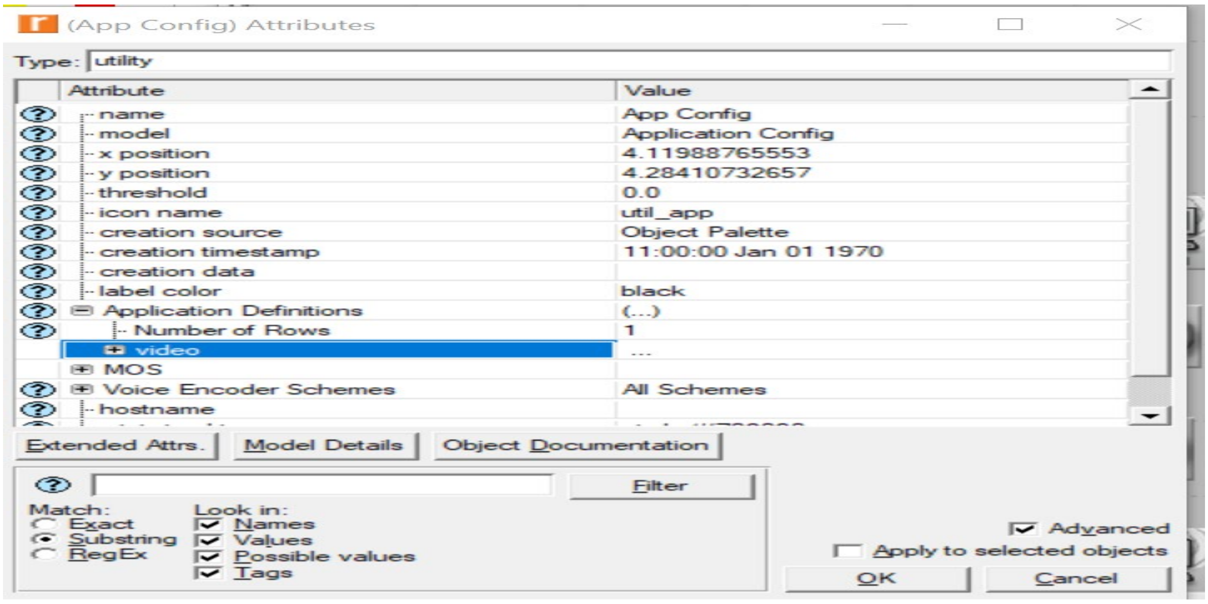


Fig. 3.8 Application configuration in the OPNET for the private UAV network

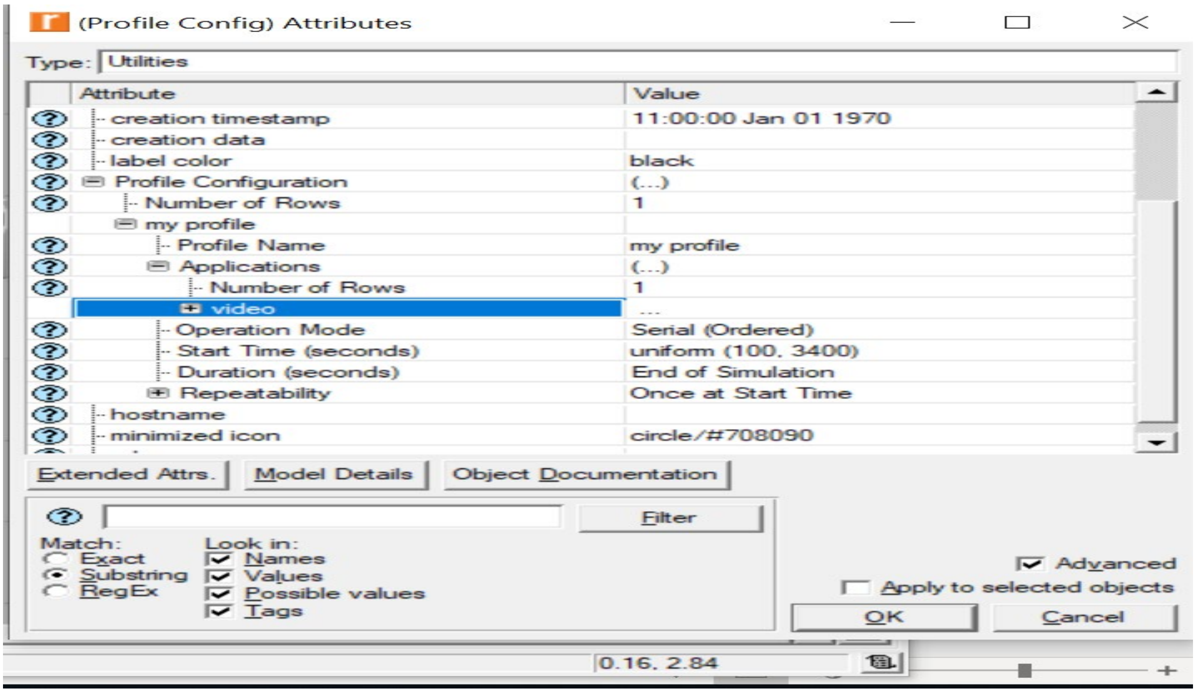


Fig. 3.9 Profile configuration in the OPNET for the private UAV network

We configured a custom profile for all the supported applications and assigned it a name as 'my profile'. We then added all the applications one by one which were created in the application configuration model for this profile configuration. Once we had the profile configuration ready, we configured each sensor device in this network for supporting this profile configuration. We also configured the wireless LAN server (ground station in the network) to support all the profiles. After these steps had been taken, our first private UAV network arrangement setup in OPNET was ready for the simulation. Before discussion of the simulation and results for the communication of this private UAV network, we are going to explain the network set up for the second private UAV network in OPNET. In this network setup, all sensor devices (Wi-Fi cameras), Wireless LAN router (WAP) and the wireless LAN server (ground station) were the same as for the first private UAV network setup. We also used the same application configuration and profile configuration setups for this network. The major change in this private UAV network setup was the position of each UAV and the communication path used for the UAV to ground station communication. These wireless LAN router components were configured in the OPNET in such a way that each UAV had a direct path or path through the intermediate UAVs to the ground station in this UAV network. The setup of the second private UAV network in the OPNET simulator is shown in Fig. 3.10.

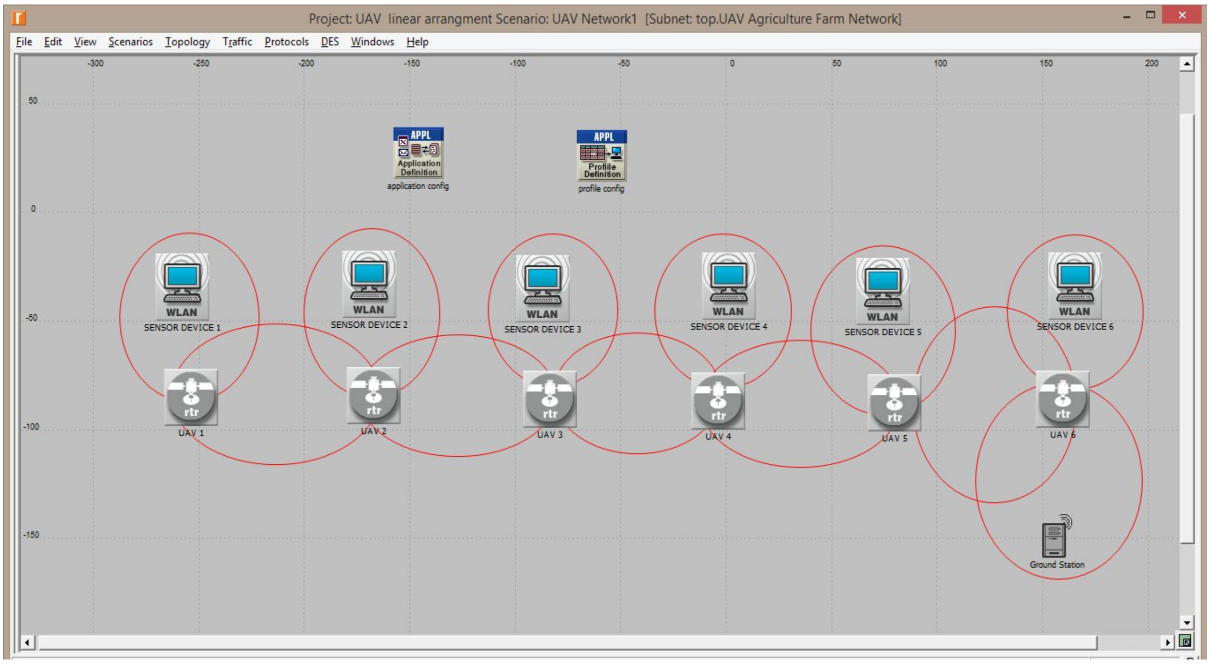


Fig. 3.10 The arrangement of UAVs in the second private UAV network

We can see from Fig. 3.10 that UAVs are connected in tandem, and the application configuration and profile configuration models are also used for this network setup in OPNET. The server, which acted as a ground station, received data from all UAVs in the network. In the next section, we discuss the simulation process and the corresponding results obtained from these two private UAV networks during the simulation.

3.5 Simulation and results analysis for the UAV communication in the private UAV networks

In the previous section, the setup of the two private networks in the OPNET is discussed. We are now in a position to explain the simulation for these private UAV networks and discuss the results of these simulations. The OPNET simulator provides the facility to monitor different statistics of simulation to verify the communication between devices in a network. We selected the individual node statistics for each node in the private UAV network, as shown in Fig. 3.11. As the wireless LAN workstation acted as a Wi-Fi camera for each UAV to generate the video data traffic and it was a wireless client of the UAV, therefore the client traffic was selected to test the communications in this network.

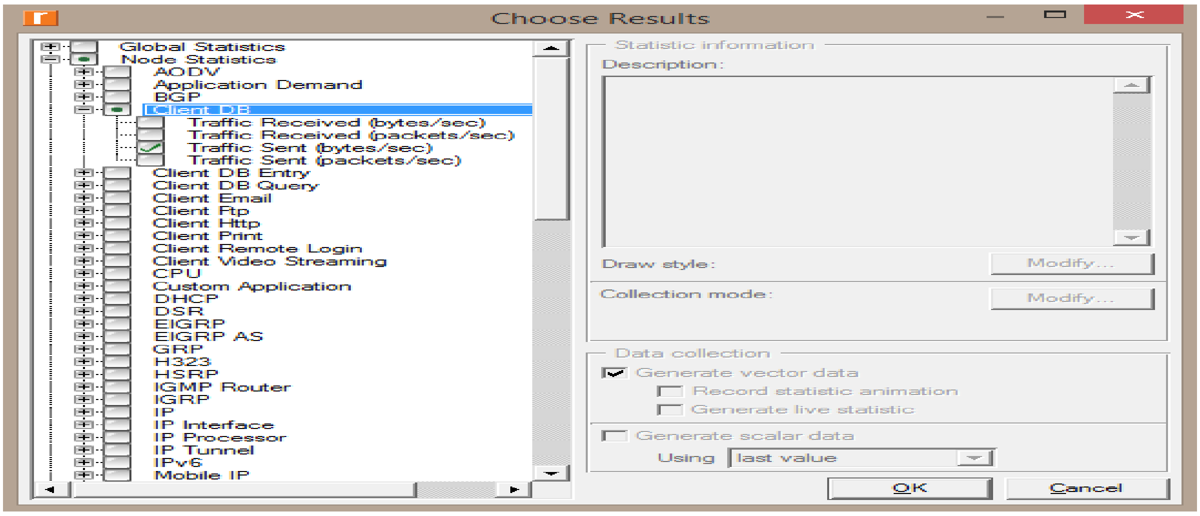


Fig. 3.11 Traffic generated statistics for each UAV

Similar to each node's statistics, we also chose the statistics for the server, which acts as a ground station in our simulation environment. The ground station statistics selected in the OPNET simulator for this private UAV network are presented in Fig. 3.12.

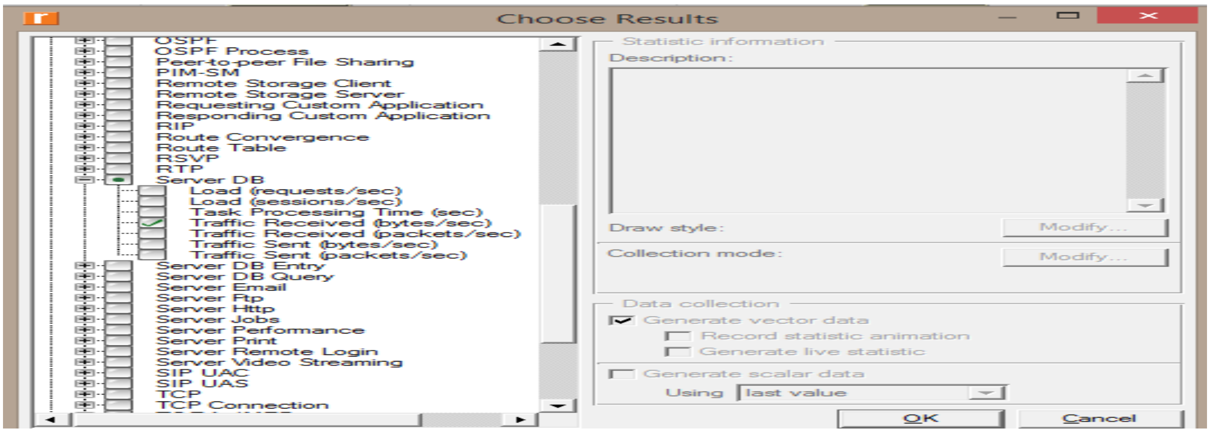


Fig. 3.12 Traffic received statistics at the ground station

As shown in Fig. 3.12, we selected the traffic received on the server in this private UAV network. Once the required statistics in the OPNET simulator had been selected for all of the components, the next step was to run the simulation. Based on the communication in both private UAV networks, whatever video data traffic generated by each UAV should have been received at the ground station. Therefore, the ground station received the data, which should have been equivalent to the addition of all data from the individual UAVs. We ran the simulation in the OPNET simulator, and it was successfully completed without any error; we then viewed the results to examine the traffic transfer from one UAV to another. The simulation of our first private UAV network worked as expected for the communication between UAVs and UAV to the ground station. Moreover, each UAV generated video data and transmitted it to the ground station. Fig. 3.13 presents the traffic generated by UAV 1 during the simulation process.

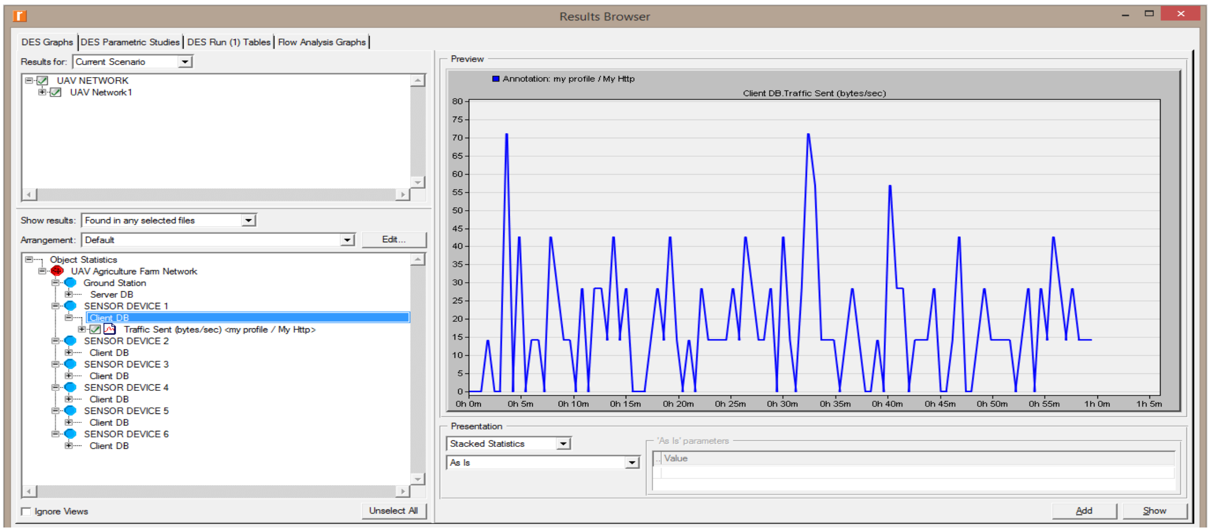


Fig. 3.13 The traffic generated by UAV 1 in the private UAV network

Correspondingly, we viewed the results for each UAV in the simulation to consider the traffic they generated within the network. We also assessed the traffic received at the ground station, which amounted to the sum of the traffic from UAV 1 to UAV 6. Fig. 3.14 presents the traffic received by the ground station in our first private network.

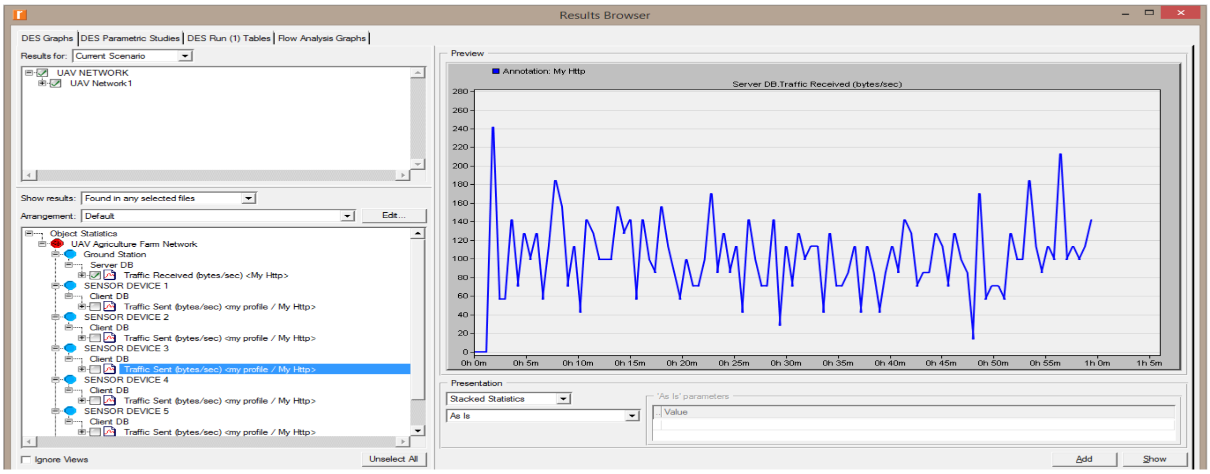


Fig. 3.14 Traffic received at a ground station

Accordingly, two graphs which show the traffic generated by each UAV and the traffic received by the ground station are presented in this section (see Fig. 3.15 and Fig. 3.16). The stacked statistics for end-to-end communication in the private UAV network are shown in Fig.

3.15. Overlaid statistics were also used to obtain a clearer picture of the end-to-end communication in this private UAV network. Fig. 3.16 shows the overlaid statistics for the first private UAV network.

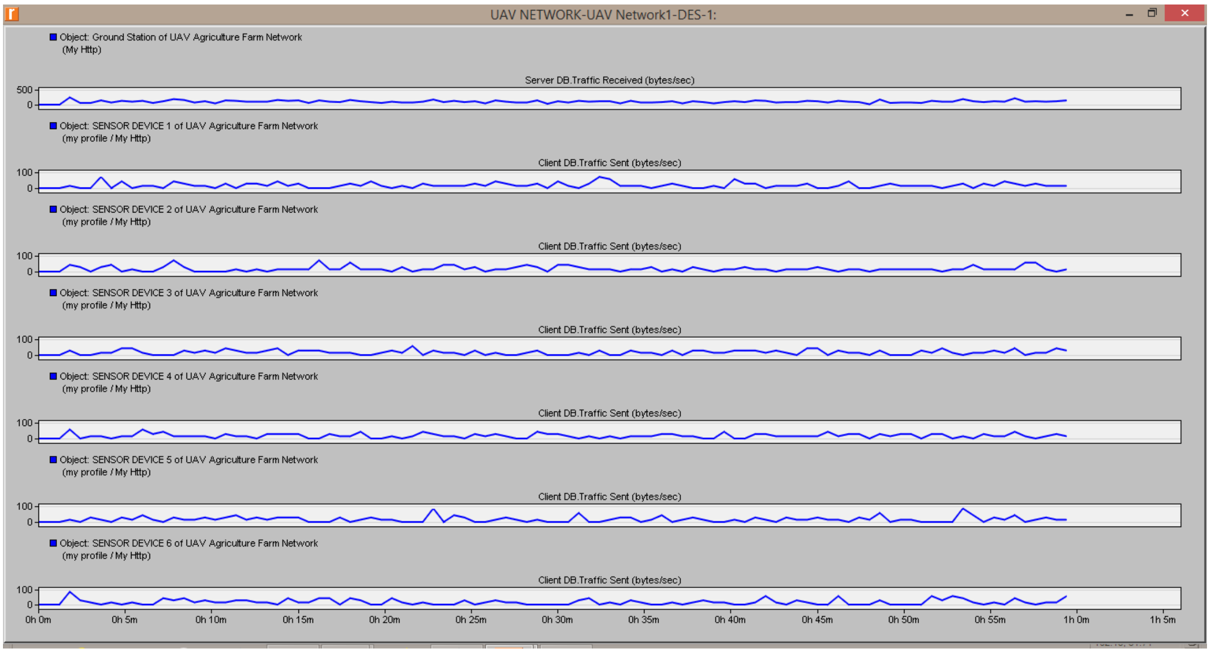


Fig. 3.15 Stacked statistics for end-to-end communication in the private UAV network

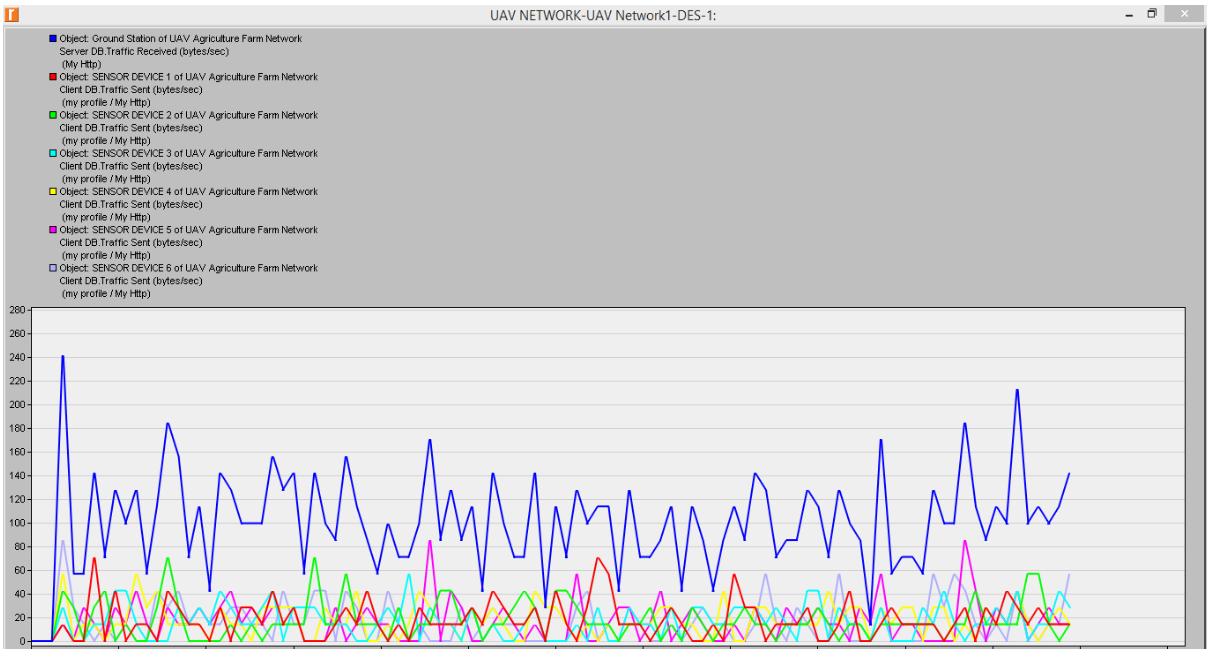


Fig. 3.16 Overlaid statistics for end-to-end communication

We performed the same simulation step to test the second private UAV network setup. The simulation results of the second private UAV network were exactly the same as for the first private UAV network.

Overall, this simulation process gave us good results for the communication between UAVs in a private UAV network. As such, we were ready to physically implement a private UAV network testbed to test these communications with the hardware components bought through the research budget. We decided to use switching to implement the communication between UAVs and UAV to the ground station for this private UAV network as routing was not an appropriate solution for communication. The switching for communication between wireless devices has many advantages over routing, and they are discussed in the next section

3.6 Simulation study with increased number of UAV nodes and different altitudes of UAVs

In the first simulation with ten UAV nodes, all the UAV nodes were placed at the altitude of 100m as shown in Fig 3.17. Altitude configuration of this simulation is also shown in Fig. 3.17.

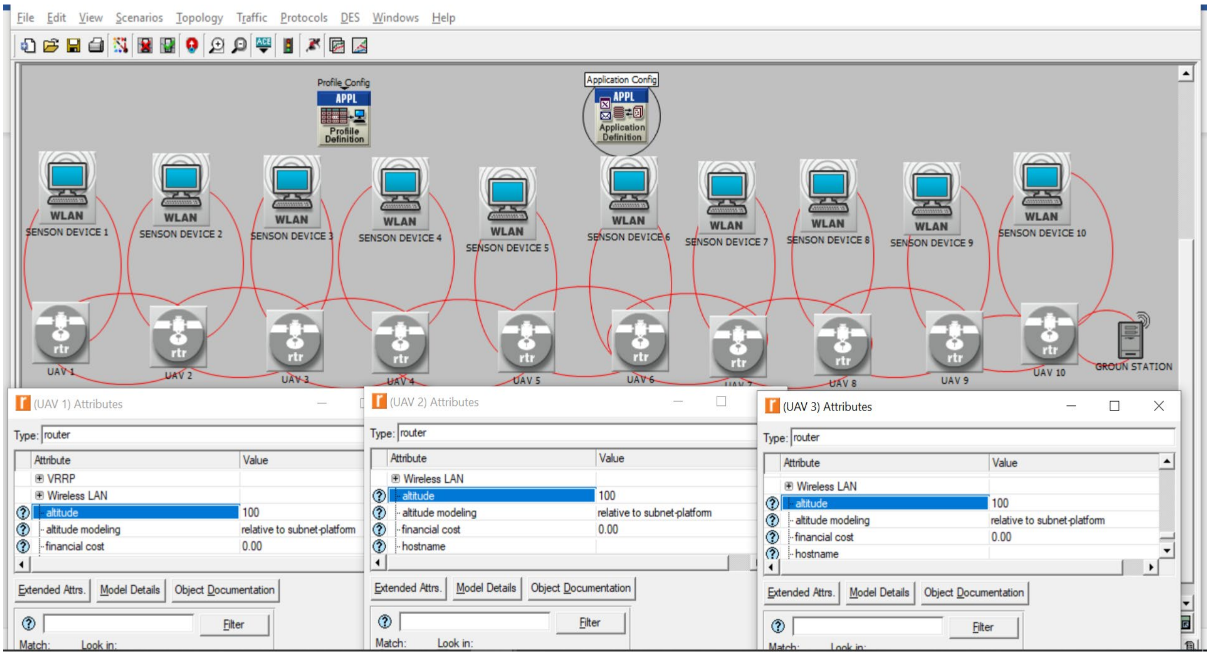


Fig. 3.17 Private UAV network with ten UAVs at 100m altitude

We selected video data transmission from UAV 1 to the ground station in the OPNET simulator and ran the simulation. Fig. 3.18 shows the video data transmission from UAV 1 and the video data received at the ground station. The ground station received the same video data as that was transmitted by the UAV 1 as shown in Fig 3.18. This simulation result proves the accurate data transmission with increased number of UAVs in the network.

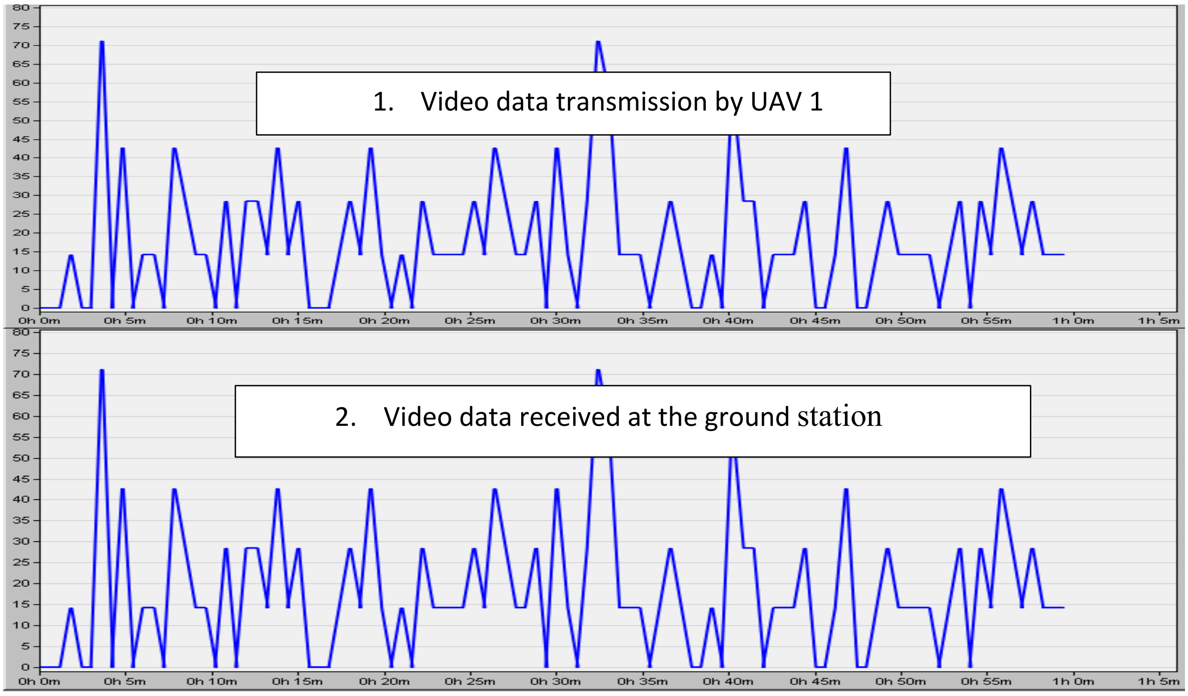


Figure 3.18 Video data transmission in the private UAV network with the same altitude

In the second simulation, we changed the altitude of the second UAV node to 150m and third UAV node to 200m and the rest of the UAVs were on the same altitude of 100m. Altitude configuration of this simulation is shown in Fig. 3.19.

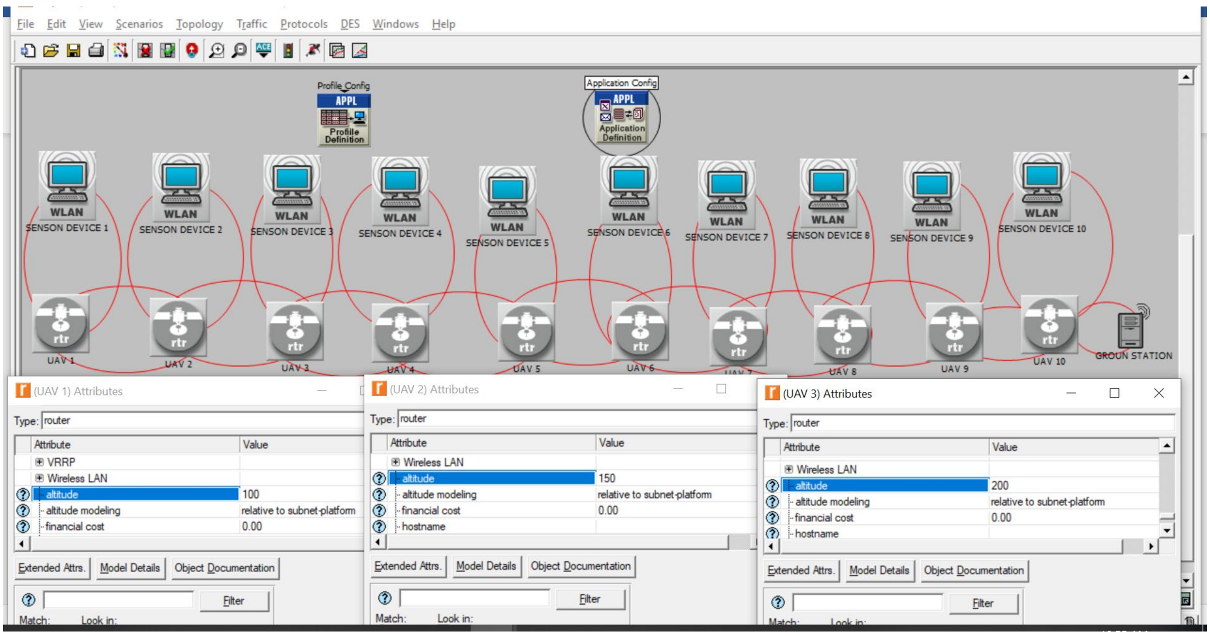


Figure 3.19 Ten UAVs with different altitude

We repeated the simulation and the results are shown in Fig. 4.24.

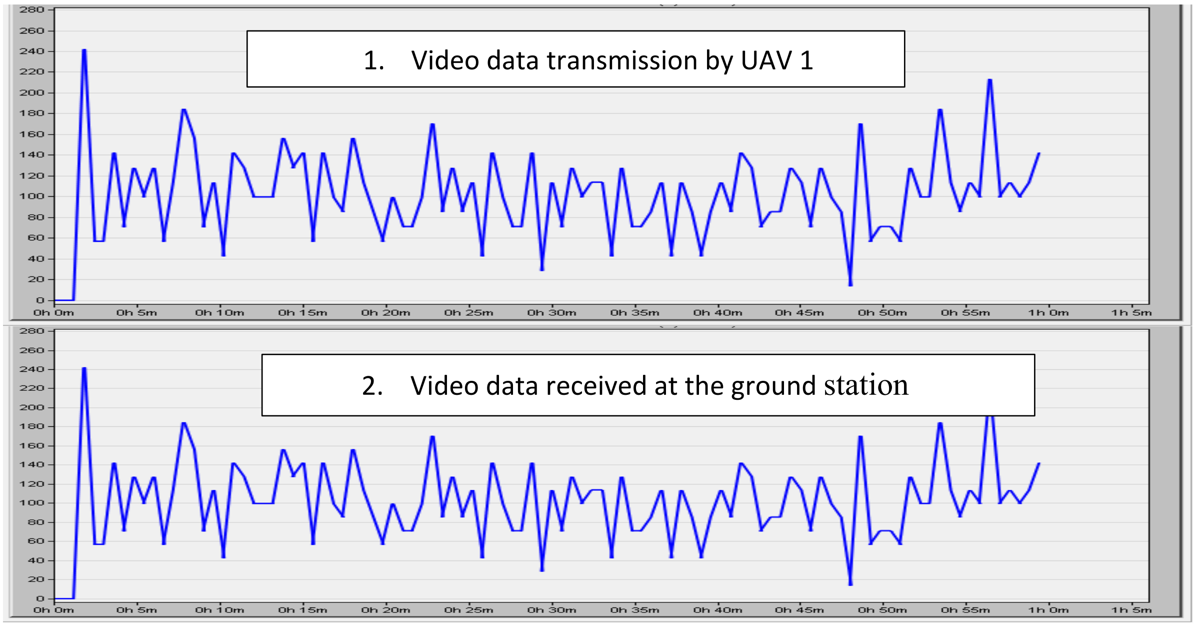


Figure 3.20. Video data transmission in the private UAV network with different altitude

The ground station received the same video data as that was transmitted by the UAV 1 as shown in Fig 3.20. These simulation results (Fig. 3.18 and Fig. 3.20) prove that there is no effect of the altitude change of UAVs on the communication as long as they are in the 100m communication range of each other.

3.7 Routing versus switching in the private UAV network

In a private UAV network, a UAV transmits its data to the other UAVs; this data is encapsulated into the Wi-Fi frame (that is, having the data, header and trailers) and then this Wi-Fi frame broadcasts over the wireless medium. UAVs, those are in the communication range of the sender UAV, receives this frame; the question here is how a UAV detects which is the is the next neighbour node that can accept this data in the network. A UAV can use two approaches, either routing or switching, to find the next hop in the network. Routing is a process for path selection between communication devices within a network or across multiple networks. Whereas switching is used to select the next-hop node for communication within a network. The routing is implemented in the routers, and it is a software-based process. On the other hand, the switching is a hardware-based approach implemented with the switch.

In UAV networks, these two approaches can be used for communication between UAVs and UAV to the ground station. The routing between UAVs can be implemented through the routing protocols discussed in Chapter 2. However, switching can be implemented by assigning the MAC hardware addresses of the neighbour nodes at each UAV in the network. As we discussed in Chapter 2, the routing protocols proposed for the FANET or have been borrowed from the MANET or VANET, are not suitable for the private UAV network due to several reasons. First, FANET has a large number of UAVs, and they belong to different users. The routing can be implemented in the high node density FANET network as each UAV can find a neighbour to route its data. However, in the private UAV network with relatively fewer UAV nodes compared to the FANET, finding the route with the routing protocols is a difficult task. Second, if we want to implement routing in the private UAV network, we have to deploy many UAVs in a small geographical area, and it would increase the cost of the network and would not be affordable for a single person. Hence, routing is not an appropriate solution to setup the communication between UAVs and UAV to the ground station in a private UAV network due to abovementioned reasons. Therefore, we implemented switching through UAV nodes to communicate one UAV to another and UAV to the ground station for this private UAV network.

Switching instead of routing in the private UAV network has many advantages. It is faster than routing, as a switch can directly send the data to the devices through the hardware address rather than spreading it to across every device in the network. Another advantage of switching over routing as it is implemented with the hardware of the WAP connected with each UAV; therefore, there is less overhead and fewer packet losses in transmission of data between UAVs. Switching is also better than routing in the private UAV network as we can easily add any additional network devices (that is, Wi-Fi camera, a monitor) within this network. Hence, switching is the best option for path selection and communication between UAVs, in a private UAV network and it can be implemented in this network through a switch by knowing the MAC hardware address of all of the neighbour UAV nodes. To add this functionality to the UAV node, we mounted additional configurable WAPs to each UAV and added the hardware addresses of its neighbour nodes. The detail of adding the hardware address into each WAP is explained later in this chapter. It should be noted that switching is implemented in this private UAV network with the Spanning Tree Protocols (STP). The main purpose of STP in the network is to avoid a redundant path that can create a loop in the network. Consequently, we enabled the STP protocol in each UAV's WAP to avoid the loops in the private UAV network.

To understand the communication in the private UAV network through switching, let us consider a situation where a farmer uses this private UAV network to monitor his farm from one ground station. We are assuming that this farm has a length of one kilometer and is in a rectangular shape. We are also considering that the width of the farm is not too big, and it can be easily captured by the UAV's cameras. A UAV has a Wi-Fi range of 100 meters. Therefore, to cover this one-kilometer agricultural farmland, ten UAVs are required to form a private UAV network. Each UAV in this network has an additional configurable WAP that acts as a switch. Therefore, switching was implemented in this network by assigning the MAC hardware addresses of neighbours UAV nodes to each UAV node through the WAPs to establish the communication between UAVs. All the UAVs are controlled via a single ground station in this network and fly according to the control commands received from the ground station or other UAV. A path always exists between the ground station and each UAV through these control commands in this network. Once each UAV in this network is in their specific position, they send their video data from one UAV to another, and finally, video data from all UAVs are received at a ground station. Hence, a farmer can monitor his farm from one location through this private UAV network implemented through switching technique.

Overall, there are lots of challenges that we faced during the implementation of this private UAV network to test the communication. In the next section, implementation of the practical testbed to verify the communication between UAVs and UAV to the ground station is presented.

3.8 Experimental testbed for the Storm 4 Mini private UAV network for communication verification

A private UAV network with six UAVs connected together is discussed in Section 3.2 and verification of the communication between UAVs and UAV to the ground station for this network (which was conducted through the OPNET simulator in this study) is presented in Sections 3.4 and 3.5. In this section, the design and implementation of an experimental testbed of the private UAV network with six Storm 4 UAVs are documented to verify the communication between UAVs and UAV to the ground station. In setting up the practical testbeds to test these communications in the private UAV network, we encountered many challenges. The most difficult problem in setting up an experimental UAV testbed network was concerned with the selection of the correct UAV model. None of the UAVs available on the market was able to communicate with other UAVs and, therefore, the transfer of data from one UAV to another was not possible.

At this point in the study, it was clear the UAV selected for our experimental testbed would always need to communicate with the help of an additional configurable WAP. We decided to select a UAV of sufficient payload capacity capable of carrying a wireless IP camera, WAP, and rechargeable battery bank to power the devices. Rather than using the built-in UAV transceivers, we used the UAV-mounted WAPs for communication between them.

Selection of the Storm 4 Mini UAV was finalised due to its high payload capabilities and cost-efficiency. There was another issue with this UAV; that is, it did not have a built-in camera. The Storm 4 Mini is controlled by a remote controller. To make it compatible with the experimental testbed, we initially used wireless IP cameras; however, since they operated in the 2.4-GHz band, there was interference from the UAV remote controllers. Therefore, we decided to replace the IP cameras with iPhones, which operate in the 5-GHz band. In the final arrangement of this practical testbed, the control signals were transmitted to the UAV through the 2.4-GHz band ground controller, which were the analog signals, and each UAV was controlled by its own remote controller. We tested communication in this network by transmitting the video signals from one UAV to another and UAV to the ground station over the 5-GHz band that used digital Wi-Fi signals. We faced another problem to fly this UAV, that is, it was not stable when flying. Moreover, the additional weight of the WAP, power bank and IP cameras made flying these UAVs a more complicated task.

Before going into the finer details of the experimental testbed of the Strome 4 Mini private UAV network, we first give some details about the UAV node used in this network. From Fig. 3.21, we can see that the Storm 4 Mini UAV has WAP, power bank and Wi-Fi camera mounted on its top, and this UAV is placed at ground level. Let us identify it as a UAV node for the Strom 4 Mini practical testbed setup.

Fig. 3.21 UAV node with the Storm 4 Mini UAV

In this private UAV network, a UAV node can send its video data to its neighbouring UAV node and, after transfer through several intermediate UAV nodes; the data are then received at a notebook computer. A neighbour UAV node of a UAV node is a node that is in the communication range of the latter. In the current study, the notebook computer acted as the ground station, but it can be replaced by a wireless router from a service provider for the video data to be sent through the Internet. WAPs in UAV nodes can also be configured to receive data from any neighbouring UAV node; however, since we could only construct six such UAV nodes due to the budget allocated for the study, we arranged them in tandem, as discussed in Section 3.4 of this chapter to implement the practical testbed private UAV networks. In the next section, we discuss the hardware components that were used in this practical testbed and then explain the corresponding network configuration with all testcases that were used to verify the communication between UAVs and UAV to the ground station.

3.8.1 Hardware component details

In this experimental testbed with Storm 4 Mini UAVs, various hardware components were required to complete the testbed setup. The first component was the Storm 4 Mini UAVs and six of these were used for the setup of this private UAV network. Accordingly, we also required six Edimax WAPs, which were mounted on top of each UAV in the network. Six iPhones were also used to work as Wi-Fi cameras. As such, we needed an energy source which could power up the WAP; therefore, six power banks were used, one for each WAP in the network. As video data would be passed from the UAVs to the ground station, we also required a notebook computer which would act as the ground station. The details of these devices are given in the following subsection.

3.7.1.1 Storm 4 Flying Platform (V2.0)

The Storm 4 Mini UAV is a lightweight remote-controlled UAV. As per the specifications outlined by the manufacturing company, this is a small-sized commercial UAV which can reach a flight duration of 8–12 minutes. The flight time of any UAV depends on the payload it carries during flight and the weather conditions in which it is flying. Therefore, the use of this particular UAV with a maximum payload corresponds to reduced flight time. This UAV is controlled by the Devo remote controller. Once this UAV has been powered up, it can be flown via the signals generated by its controller. Correspondingly, each UAV controller in this particular case is bound to the UAV by a pre-defined code, which meant that we could only control each UAV through the controller connected to it. Table 3.3 outlines several features of the Storm 4 Mini UAV (more detailed information about this particular UAV model is provided in Appendix A) [9].

Table 3.3 Storm 4 Mini UAV specification

Dimensions	490 mm x 490 mm x 140 mm
Motor to motor	450 mm
Propeller size	8" x 4.5" Carbon Fibre Propeller
Take-off weight	850 gm (max. safety 1350 gm)
Battery	11.1 V
Flight time	8–12 minutes
Max. payload	250

3.7.1.2 Edimax BR-6288 ACL

We searched extensively for a suitable WAP to use in the private UAV network. The main objective was to find a WAP which was lightweight and easy to mount on top of a UAV. Another challenge was finding a WAP which would support the WAP bridge mode and not exceed the budget allocated for the study. Therefore, the selection of a WAP was finalised with these two aspects in mind. In the practical testbed network, we used the Edimax BR-6288 ACL WAP, which can be operated on 2.4 and 5 GHz wireless networks. This Edimax device operates in five modes: router, access point (AP), range extender, Wi-Fi bridge and wireless Internet service provider (WISP). It uses the IEEE 802.11 a/b/g/n/ac standard to communicate with other Wi-Fi devices. According to the manufacturer, this WAP works at a speed of 150 Mbps. The key features of this WAP are summarised in Table 5.2 (more detailed information on this WAP is provided in Appendix E) [67].Table 3.4 Edimax WAP specification

Functions	Supports router, access point, range extender, Wi-Fi Bridge and WISP modes; up to 10 SSIDs (2.4 GHz x 5 and 5 GHz x 5) with VLAN support in the AP mode.
Hardware interface	1 x micro USB power port; an internal high gain antenna; WPS/reset button; and wireless normal-/green mode switch.
WAN	WAN protocol: PPPoE, static IP, dynamic IP, PPTP and L2TP.
Security	64/128-bit WEP, WPA, and WPA2 security.
Power adapter	DC 5V, 1.2A

3.7.1.3 Mobile IP camera

We installed the free small camera application on the iPhones used in this practical testbed, as a Wi-Fi IP camera for the UAV node. These cameras acted as video servers and could be accessed via a laptop using their IP addresses. The advantage of using these iPhone cameras was that they were lightweight and did not require additional battery power as they contained built-in batteries. Since iPhones are expensive devices and were not covered in our research budget, we used our personal iPhones to test the communication between UAVs and UAV to the ground station in this network.

3.7.1.4 Power bank for WAP

The next hardware component used in this private UAV network was the power bank. There was plenty of power banks available on the market, and the objective was to select a lightweight power bank capable of generating enough power for the WAP (that is, 5 V). We finalised our selection based on these requirements.

3.3.1.5 Notebook laptop (ground station)

The final component in this private UAV network was the notebook laptop which worked as the ground station. As the whole private UAV network was a wireless network, we required a laptop which could connect with this network via a wireless mode. All modern notebook laptops contain Wi-Fi adapters which can be connected with any Wi-Fi device. Similar to the use of our personal iPhones in this practical testbed, we used our own laptop to act as the ground station in the network.

3.8.2 Configuration of devices

All the required devices to form this Storm 4 Mini private UAV network were discussed Section 3.7.1. However, these devices need to be configured to use in the implementation of a practical testbed. In this section, we explain the configuration of these devices one by one with all the steps that followed to configure these devices. First, we give the details about the steps used for the configuration of Edimax WAP and then explain about the configuration for the smartphone IP cameras as well as the notebook computer (ground station) for the Storm 4 Mini private UAV network testbed.

3.8.2.1 WAP Configuration

For the Storm 4 Mini private UAV network practical testbed, we used switching technique to enable communication between UAVs and UAV to the ground station through their MAC hardware address and, therefore, each WAP was configured in the Wireless Distribution System (WDS) mode. Before an explanation on the configuration of the WAP, the following section introduces the WDS bridge mode of WAP.

The WDS bridge mode of WAP

The WDS bridge mode of WAP is a special type of configuration used to enable one WAP to communicate with another WAP in a wireless medium. The WDS configuration is a MAC hardware address-based configuration in which the MAC address of all the neighbour WAPs are added to the central WAP. For example, if we have three WAPs connected in tandem with the WDS configuration, then the first WAP has the MAC hardware address of WAP 2, WAP 2 stores the MAC hardware addresses of WAP 1 and WAP 3 and WAP 3 has the MAC address of WAP 2. The WDS bridge operates with two modes: wireless bridge mode and wireless repeating mode. In the private UAV network testbed, each WAP was configured in the WDS bridge mode and, therefore, the two WAPs could communicate directly with each other through the MAC hardware addresses. The basic steps which must be followed to set up WDS WAP-to-WAP communication are given in order below:

- Assign a static IP address to each WAP;
- Set a common channel number to each WAP;
- Assign a common or different SSID to each WAP;
- Place all WAPs in such a way that any two neighbour's WAPs should be within communication range of each other;
- Record the MAC address of each WAP in the network; and
- Assign the neighbour's MAC addresses to each WAP.

We followed these abovementioned steps to configure each WAP for the private UAV network in this study. As mentioned earlier, we used the Edimax WAP for the setup of the Storm 4 Mini private UAV network. This WAP operates on two-frequency bands of 2.4 and 5 GHz. In our early experiments, we configured the WAP on the 2.4-GHz wireless network;

however, there was interference from the Storm 4 Mini UAV motor driver signals and, therefore, we had to use the 5-GHz band. The default IP address of each WAP supplied by the manufacturer was 192.168.2.1. We changed the IP address of each WAP for our private UAV network; therefore, the first step in the WAP configuration was to change the IP address of each WAP (as shown in Fig. 3.22).

We changed the IP address of each WAP but kept them on the same subnet. The new IP addresses of the WAPs were set from 192.168.2.2 to 192.168.2.7. We saved the setting and rebooted each WAP to make sure that new IP addresses were assigned to each of them. Fig. 3.22 shows the new IP address for our first WAP. We also enabled the Dynamic Host Configuration Protocol (DHCP) server in this WAP to automatically assign the IP address of the Wi-Fi iPhone camera. The wireless spanning tree was also enabled to avoid a loop in the network.

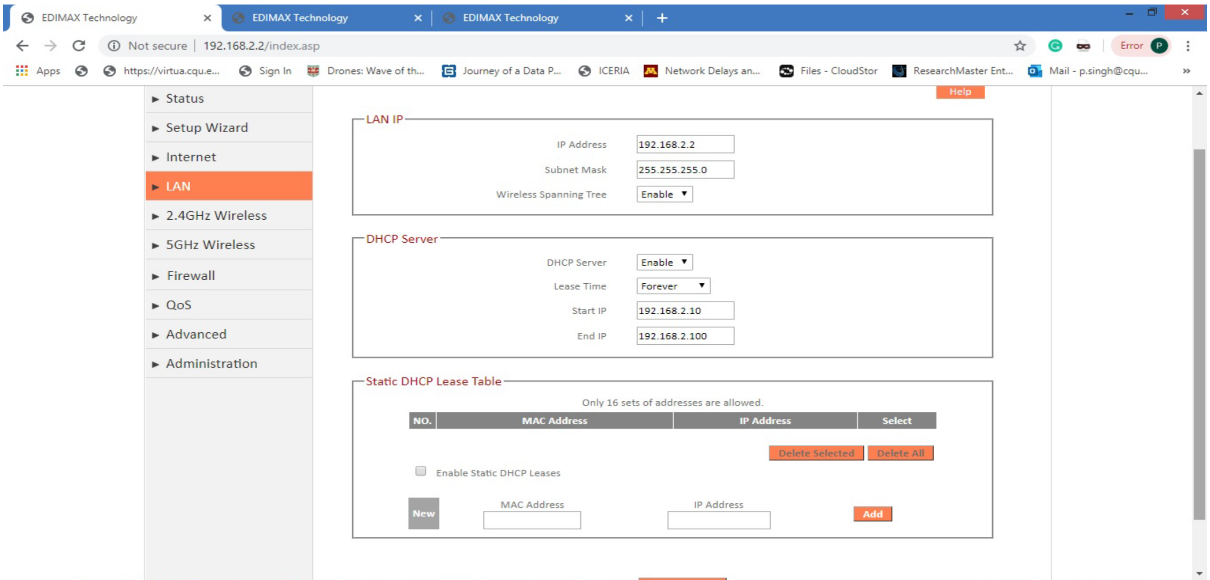


Fig. 3.22 Assigning the IP address to WAP 1

The next step in the WAP configuration was to set up the wireless WDS bridge mode for each WAP. Since the Edimax WAP operates in five different modes and we were only interested in the AP mode (and, furthermore, as it also has two different AP modes: normal AP and AP bridge WDS modes), we selected the AP bridge WDS mode on each WAP and rebooted it after saving the settings. Our plan was to enable communication between the two UAVs using their MAC hardware addresses for data transmission, which was only possible in the bridge WDS mode. After rebooting the device, we checked and found that the AP mode was set to the AP WDS bridge mode. We changed some other configurations in the WDS mode of WAPs and modified them one by one. We gave a common SSID (that is, ‘WAP1’) to each WAP in the configuration. As discussed earlier, the WDS mode only operates on a common communication channel. We fixed the common channel number ‘36’ to each WAP.

The next step in this WAP configuration was to assign the MAC addresses of the neighbouring WAPs to each middle WAP; we noted the MAC address of the 5-GHz wireless device as 74:DA:38:A5: FF:45 (MAC address of the first WAP shown in Fig. 3.23), which was assigned to this WAP by the manufacturer. In the Edimax WAP WDS configuration, a maximum of four MAC addresses can be stored. In each WAP, we saved the MAC addresses according to our network setup plan. Based on our experimental testbed network, UAV 1 had the MAC address of UAV 2, UAV 2 had the MAC address of both UAV 1 and 3, and so on.

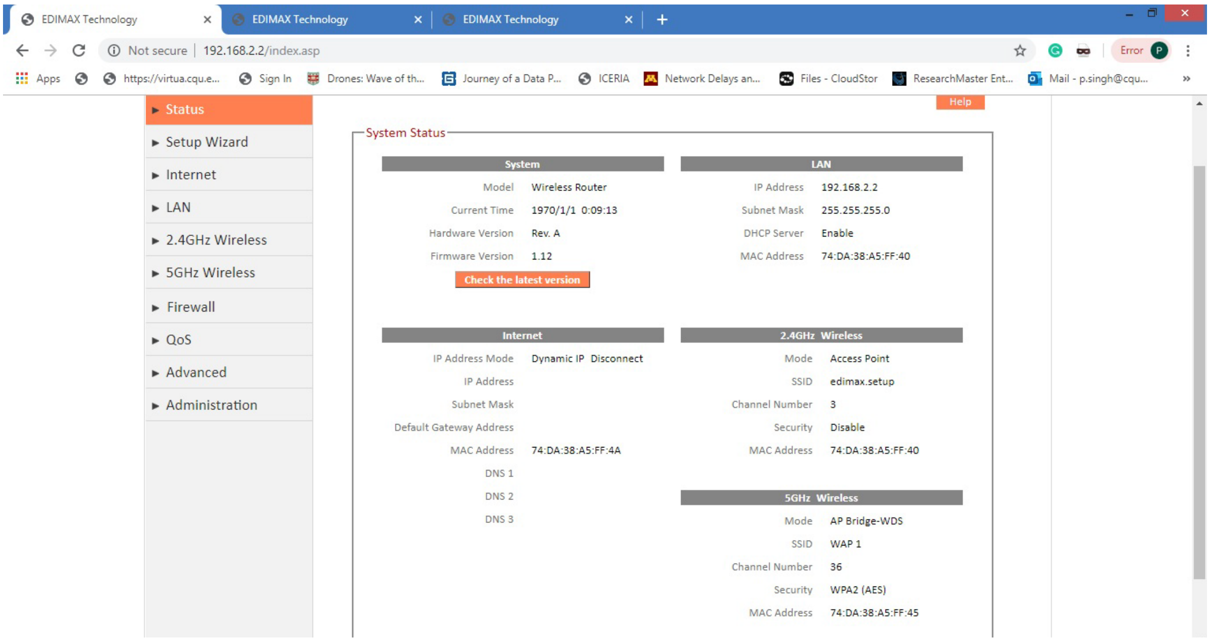


Fig. 3.23 MAC address of WAP 1

The last step in this configuration was to set up the security to each WAP. We chose the WPA 2 security and set the password of each WAP to ‘Testnetwork’. We saved this configuration setting and rebooted the WAP to ensure the changes were successfully implemented. Fig. 3.24 shows the bridge mode setting of the first WAP.

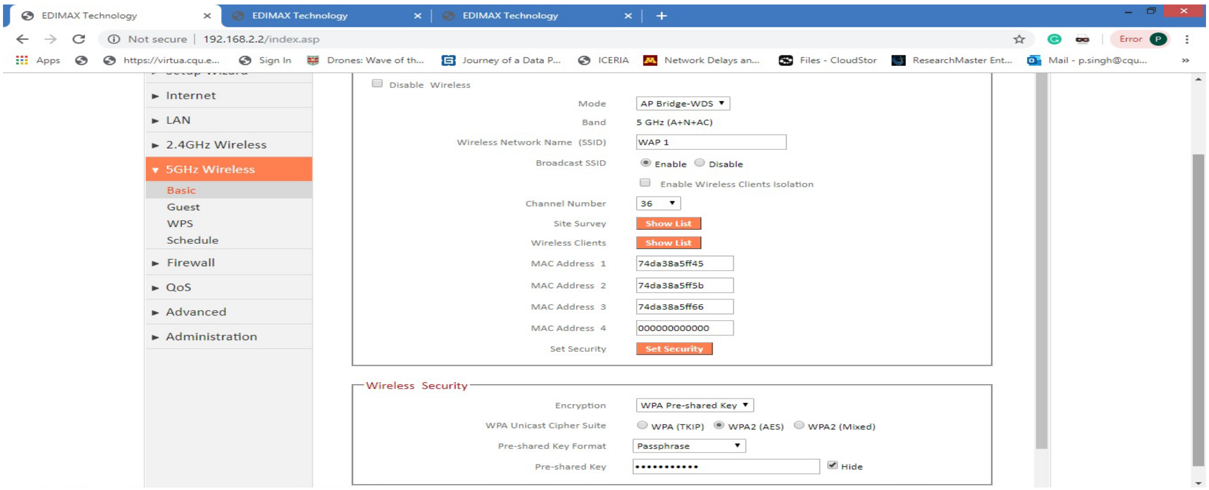


Fig. 3.24 WDS bridge mode of the Edimax WAP

Table 3.5 WAPs configuration

IP addresses	192.168.2.2 to 192.168.2.7
Wireless	5-GHz network
SSID	WAP 1
Security	WPA2
Password	Testnetwork
Mode	AP Bridge WDS
Common channel	36
MAC address	Each UAV had the MAC address of the previous and next UAV
Wireless spanning tree	Enabled
Transmitting power	15 per cent

The default communication range of the Edimax WAP was approximately 100 meters. Accordingly, we needed to test the communication between UAVs and UAV to the ground station through these WAPs in this network with six of the UAVs in our university campus. However, this was not possible with the full communication range of the WAP due to limitation of space in the university campus. Therefore, we changed the communication range in the advanced settings of this WAP and dropped the transmitting range to 15 percent. Our first WAP was then ready to be used in the network. We configured each WAP according to the process outlined above. The final configuration of the WAPs is shown in Table 3.5 (below).

3.8.2.2 Smartphone camera configuration

We used a smartphone iPhone 5 to act as a Wi-Fi camera for the Storm 4 Mini private UAV network and installed a free IP camera application on it. After connecting the smartphone to the wireless network, we started the IP camera application to transform it into an IP web server. We then assigned IP addresses of the same subnet of the 5 GHz UAV network to the smartphones. The configurations of the smartphones are given in Table 3.6.

Table 3.6 Smartphone IP camera configuration

Smartphone	iPhone 5
IP addresses for access videos	192.168.2.10 to 192.168.2.15
Wi-Fi network	WAP 1
Password	Testnetwork
IP camera app	IP camera light free app
Web address to access live video	192.168.2.10:8081 to 192.168.2.15:8081
Resolution	640*1136 p

3.8.2.3 Ground station notebook laptop configuration

We used our notebook laptop to act as a ground station for the private UAV network testbed. We manually assigned the IP address to the laptop (that is, 192.168.2.1.) in the same subnet of the UAV network and connected our laptop to WAP 1 SSID that was assigned to each of the WAP in this private UAV network.

3.9 Communication verification in the Storm 4 Mini Private UAV Network

We implemented the experimental testbed for Storm 4 Mini private UAV network and performed various test cases to test the communication between UAVs and UAV to the ground station. A discussion on these test cases is presented in the following subsection.

3.9.1 Test Case 1: WAP network setup

For the network setup, we first configured all the WAPs by giving them different IP addresses and then operated the network at 2.4 GHz. We connected the WAPs one by one to the laptop on the same wireless network and then checked the communication using ping commands. We noted the maximum wireless range at which the laptop could access a WAP. We then set the transmitting power in each WAP to 15 per cent to reduce the wireless range. In this first experiment, each WAP was successfully configured (as shown in Fig. 3.25).

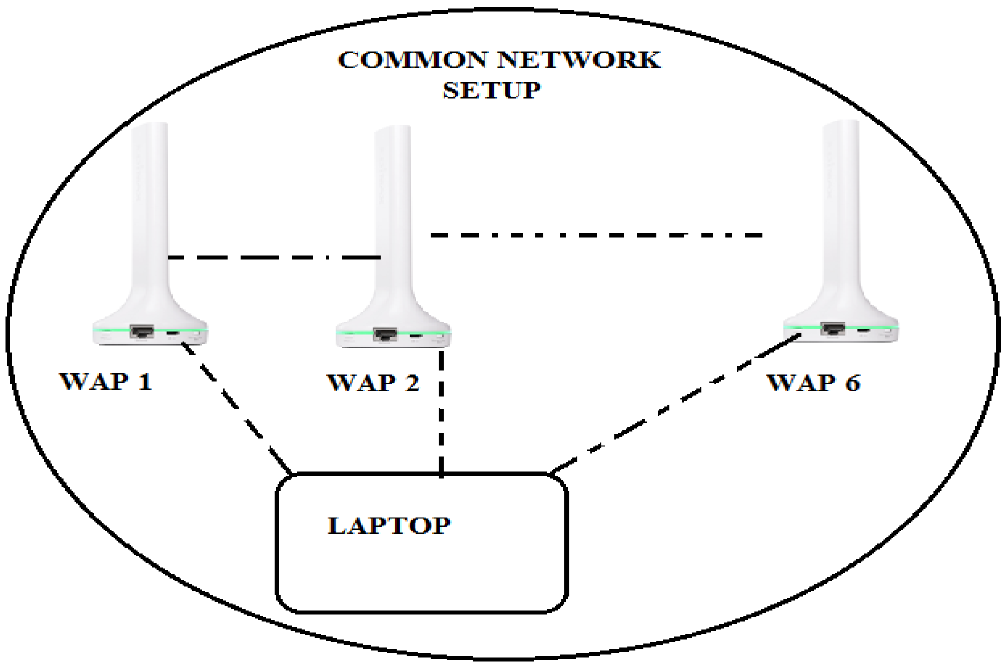


Fig. 3.25 WAP network setup

3.9.2 Test Case 2: Video streaming using WAP

In our second experiment, we connected a smartphone and operated it as a Wi-Fi IP camera. We used the free IP camera app to test the live video streaming function with the WAP at our laptop. We connected this smartphone to a WAP on the 2.4-GHz Wi-Fi network and accessed the live video on the laptop (as shown in Fig. 3.26). In this test, we were able to watch live video streaming at the laptop without any interference.

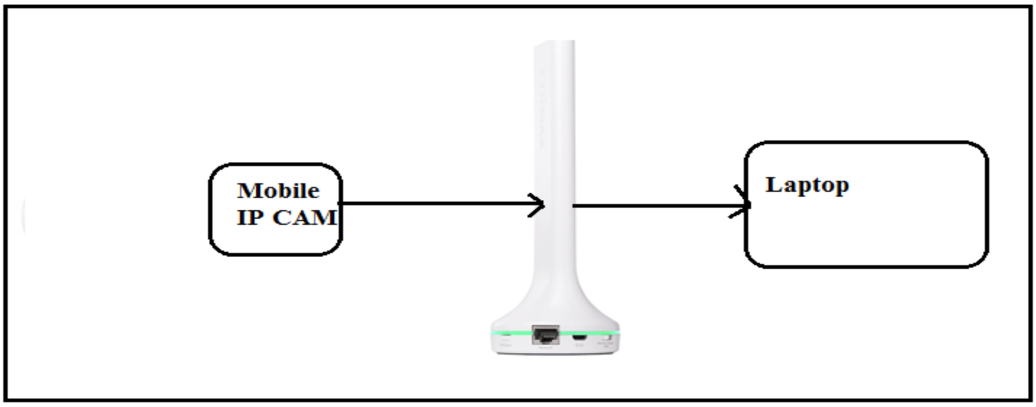


Fig. 3.26 Live video data access at a laptop using WAP.

3.9.3 Test Case 3: Video streaming with two WAPs

In the next experiment, we tied the smartphone and WAP together. As we already knew the range of the WAP, we placed another WAP in such a way that it was at the extreme of the field of the first WAP. We then placed our laptop in the field of the second WAP but not in the field of the first WAP. We configured all of the WAPs, IP cameras and the laptop with the 2.4-GHz Wi-Fi network. Following this, we tested the live video and found that it could be accessed by the laptop. The network setup for this experiment is given in Fig. 3.27.

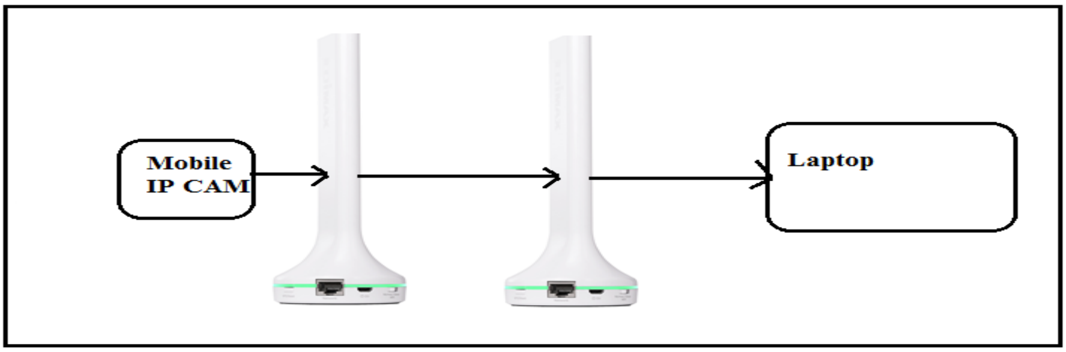


Fig. 3.27 Video data transfer using two WAPs.

3.9.4 Test Case 4: Video data transfer with UAVs

Our next experiment was with the Storm 4 Mini UAV; we mounted a WAP, a power bank and a smartphone on a Storm 4 Mini UAV. This UAV was able to fly with these devices, as the total payload of these devices did not reach the maximum payload associated with this UAV. We then powered up the WAP with the power bank and connected the smartphone to the same 2.4-GHz Wi-Fi WAP network. Next, we placed our laptop within range of the WAP and connected it to the same network. We started the mobile IP camera app and checked the live video streaming on the laptop—specifically, we received excellent video streaming in this experiment. This experiment was conducted on our university campus, and these Storm 4 Mini UAVs were very difficult to control in flying mode with the controller as they were not stable during flight. Therefore, we took off the rotors of the UAV before powering it up. Upon starting our UAV with the help of the remote controller, we found that we had lost the video signal on the laptop. The cause of the issue was determined to be too much interference, as our mobile IP camera and the controller both operate on the 2.4-GHz frequency (that is, the same frequency band as for our network). As such, we decided to transfer the video data at 5 GHz to avoid the interference from the 2.4-GHz signals to test the communication in this network.

3.9.5 Test Case 5: 5-GHz Wi-Fi network setup

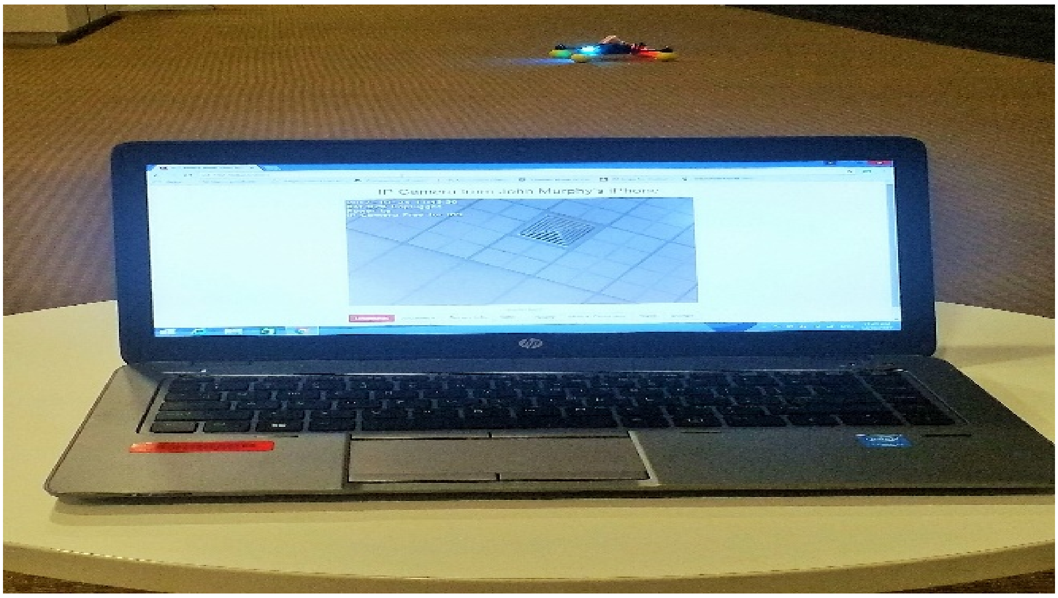


Fig. 3.28 Experimental setup for transfer of video data using an iPhone 5 on the 5-GHz network

We configured all WAPs on 5 GHz wireless network, but our smartphone did not operate on the 5 GHz network. So, we decided to use iPhone 5, which was able to work with the 5 GHz wireless network. We repeated the same experiments and this time we were successful in live video streaming as our video data was transferred through the 5 GHz wireless network, and the control signals for the UAV were at 2.4 GHz. This live video streaming is shown in Fig. 3.28.

3.9.6 Test Case 6: Transfer of video data using two UAVs and one mobile IP camera

In this test case, the second UAV node was placed within range of the first UAV node. We then powered up both UAVs and received the video data on the laptop, which was placed in the field of the second UAV. Therefore, the flow of data was received from the mobile IP camera to its WAP, which was mounted on the first UAV, and then to the second WAP, which was mounted on the second UAV. The second WAP then forwarded the data to the ground station (that is, the laptop). The setup for this test case is given in Fig. 3.29.

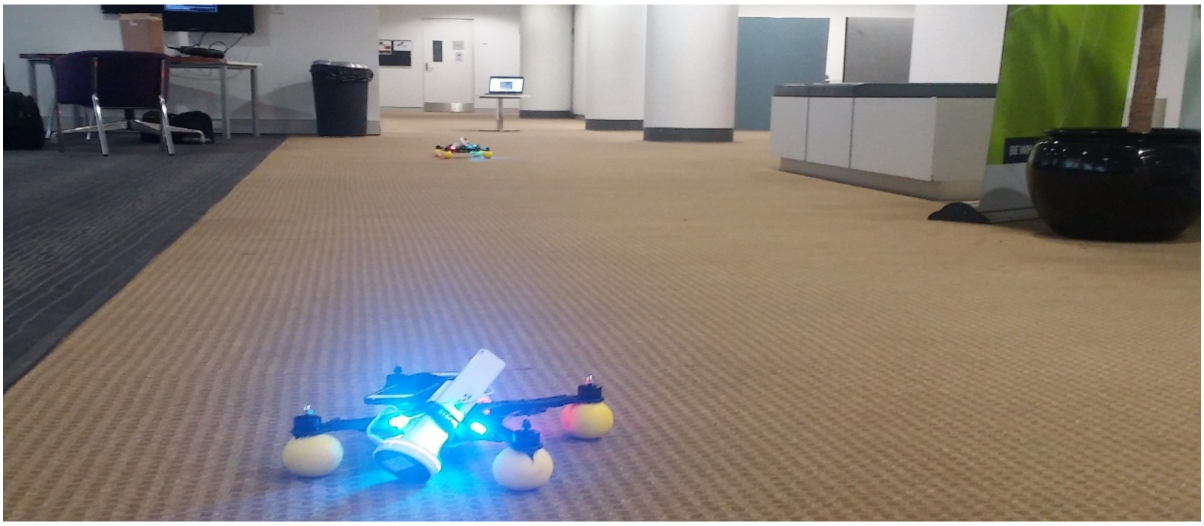


Fig. 3.29 experimental setup for transfer of video data using two UAV nodes

3.9.7 Test Case 7: Transfer of video data using two UAVs with two mobile IP cameras

Our next experiment was conducted with two mobile IP cameras. We mounted the second iPhone on the second UAV and tested it again. Our laptop (that is, the ground station) was now receiving two video streams: one from the first iPhone and one from the second. We encountered several range issues, but these were easily resolved by changing the position of the UAVs. The flow of data was then received from one UAV node to another and then to the ground station.

3.9.8 Test Case 8: Transfer of video data using six UAVs

In the final experiment, we mounted the mobiles and WAPs onto all of the UAVs. We then received video data from each UAV to the laptop after adjustments were made to the range of these UAVs. Each UAV transferred video data from its iPhone and also forwarded the video data from the previous UAV to the next through these WAPs. The experimental setup for this test case is shown in Fig. 3.30.

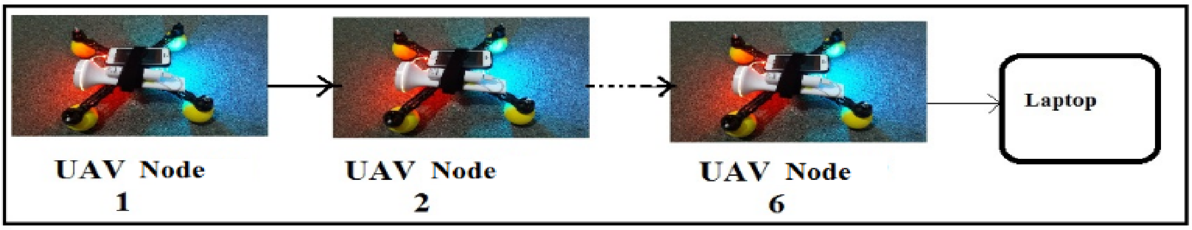


Fig. 3.30 Video streaming using six UAV nodes in tandem

Though this private UAV network testbed, we successfully implemented the communication between UAVs and UAV to the ground station at the MAC layer by switching through UAV nodes. The main purpose of this network is the fast transmission of video signals from one UAV to another and then to the ground station by switching through UAV nodes. However, some delay was noticed in these live video streaming form each UAV, and it was due to using a common communication channel between UAVs in this network setup. We analysed this issue, and a detail discussion about this video streaming delays is presented in Chapter 4.

3.9.9 Verification of the communication path

To verify the communication path in this experimental testbed, we powered off some of the WAPs (that is, 2, 3, 4, and 5) during the experiment. As soon as this occurred, the video streaming stopped, and we were unable to see the live videos. This confirmed that the video data which was being received at the laptop, passed through all of the WAPs before it reached its final destination. We also used Wireshark to confirm this communication between UAVs in this practical testbed. Wireshark is a free and open-source packet analyser tool used in a network. For this purpose, we captured the Wi-Fi frames from each of the WAPs in Wireshark by pinging the relevant WAP IP address. The details of the first three WAP Wi-Fi frames that we captured using Wireshark are given in the following subsection.

3.9.9.1 WAP 1 frame capturing

In this testbed, we accessed the first WAP from our laptop by pinging it with its IP address and, simultaneously, we captured the traffic in the Wireshark. Fig. 3.31 shows the ping request and reply message between the ground station (that is, the laptop) and the first WAP that was captured in the Wireshark. From Fig. 3.31, we can see that a request was generated by the 192.168.2.1 IP address. This IP address belonged to the laptop, which acted as the ground station. This request message was received by another IP address (that is, 192.168.2.2), which was the destination of the request; and, furthermore, this IP address was assigned to WAP 1. Therefore, it is clear that communication was established between the ground station and the first wireless access point.

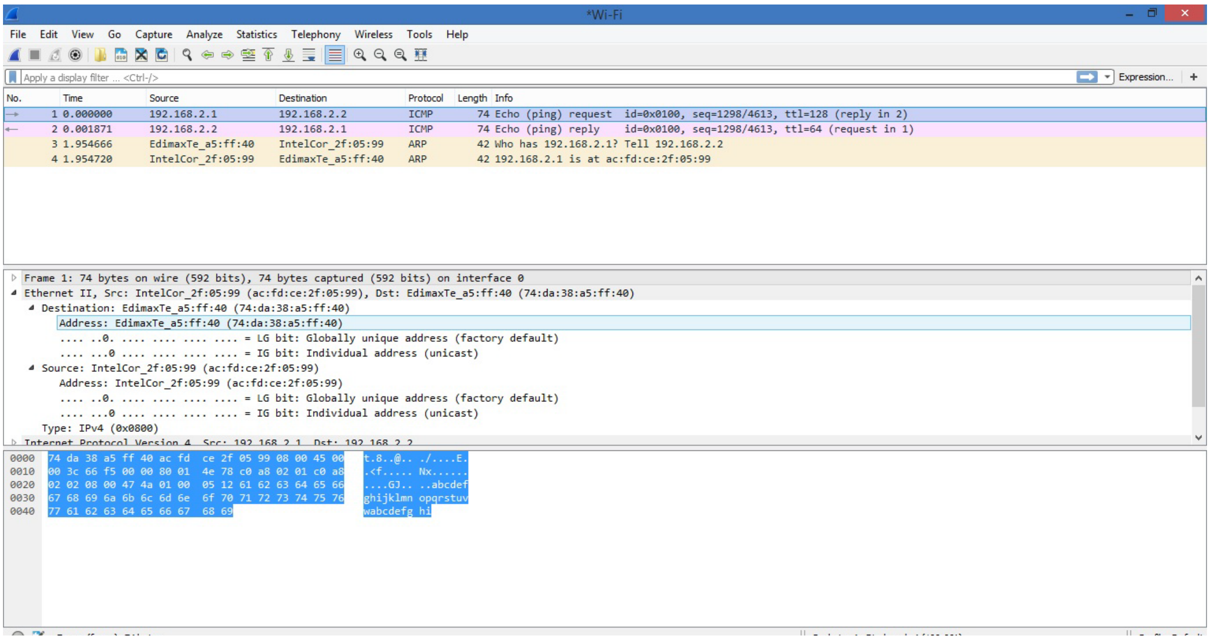


Fig. 3.31 WAP 1 frame capturing

We also checked the communication path from the camera mounted on the UAV to the ground station in this private UAV network testbed. We connected our laptop with the UAV network and accessed the live video streaming from UAV 1 by giving 192.169.2.10:8080 address (this address belongs to the iPhone Wi-Fi camera that was mounted on the first UAV) in a web browser. We then captured the incoming and outgoing frames during the video transmission with the Wireshark. Fig. 3.32 illustrates the Wireshark captured frame for the video traffic generated from IP address 192.168.2.10 to the ground station (that is, 192.168.2.1). From this Wireshark traces, we can see that the video data transferred from UAV to the ground station in the form of TCP packets. In this ping request, the source IP address was 192.168.2.1, and the destination IP address was 192.168.2.10; furthermore, in the reply message, the source IP address was 192.168.2.10, and the destination address was 192.168.2.1. Therefore, it was confirmed the live video streaming that was received from the UAV 1 to the ground station.

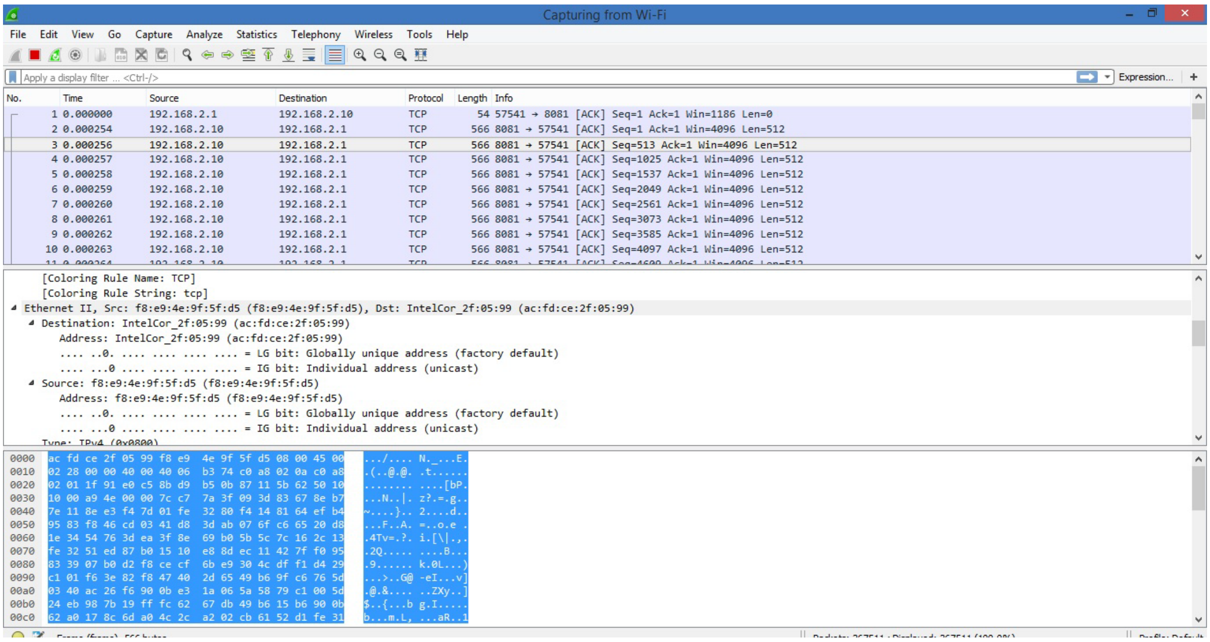


Fig. 3.32 Frame details of UAV 1 video streaming

3.9.9.2 WAP 2 frame capturing

We captured the traffic from UAV 2 in the same way as from UAV 1. The WAP IP address of UAV 2 was 192.168.2.3, and the smartphone Wi-Fi camera IP address of this UAV was 192.168.2.11. Wireshark captured frames for this WAP and smartphone camera are shown in

Fig. 3.33 and 3.34 respectively. When we sent a ping request with the IP address of the second UAV (192.168.2.2), we successfully received reply messages on the laptop which was connected with the UAV Wi-Fi network. We then accessed the live video streaming of UAV 2 in the web browser (with IP 192.168.2.11:8080). The frame captured by the Wireshark shows the transmission of video data from UAV 2 to the ground station (see Fig. 3.34).

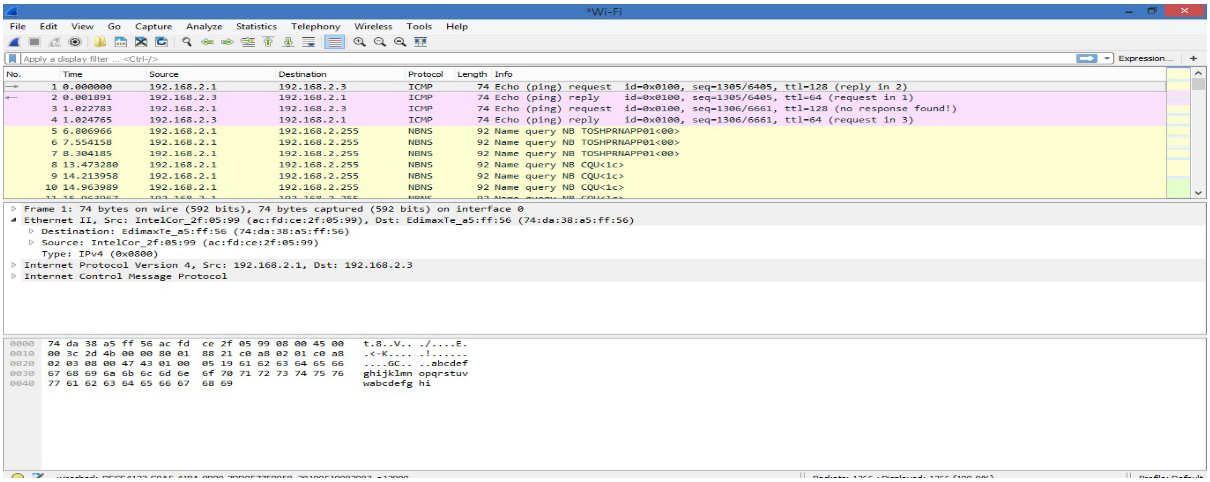


Fig. 3.33 WAP 2 frame capturing

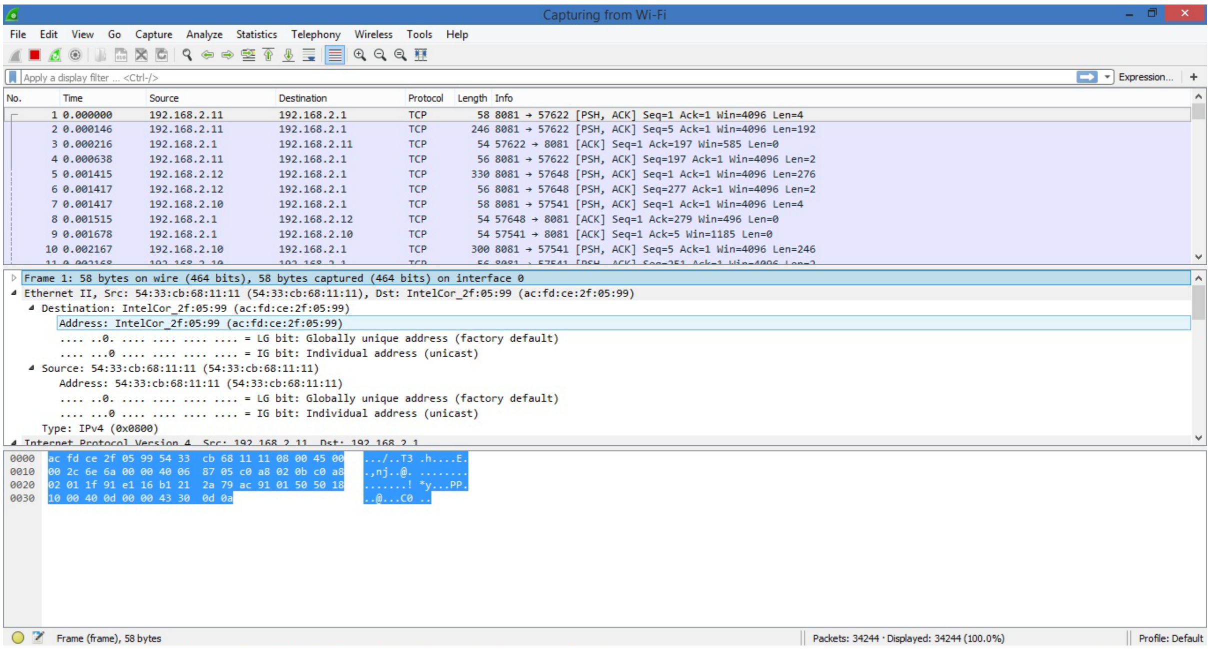


Fig. 3.34 Frame details for UAV 2 video streaming

3.9.9.3 WAP 3 frame capturing

We recorded the frames of each WAP and smartphone camera for this private UAV network. We found that the traffic moved very smoothly, and at the ground station, we received a live video streaming from each UAV. We also used Wireshark to capture the live video streaming frame from the third WAP. Fig. 3.35 shows the frames that were captured with the ping request and reply messages of UAV 3 to the ground station communication in this private UAV network.

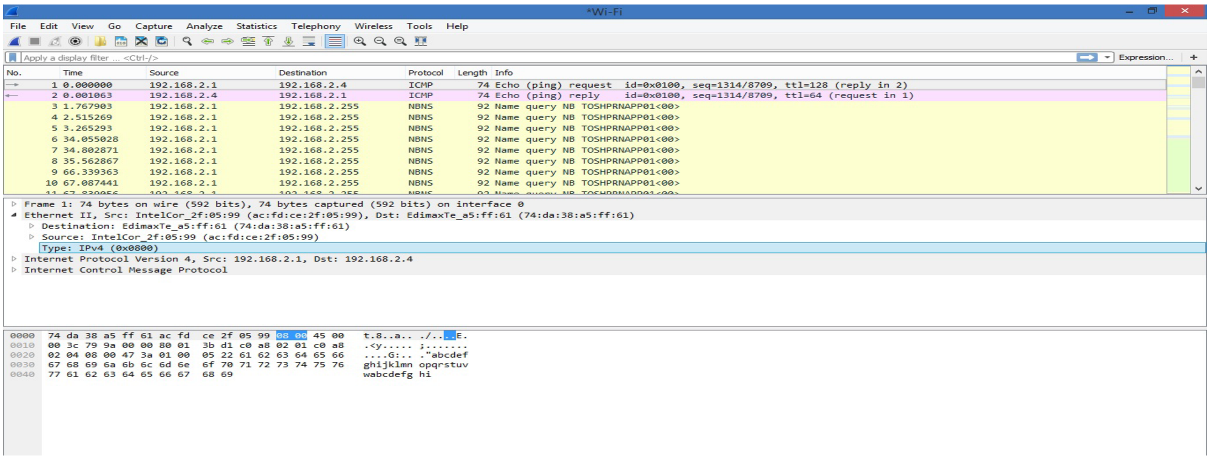


Fig. 3.35 Frame captured from WAP 3

We checked the various Wireshark frames captured during the experimental testbed to confirm the UAV to UAV and UAV to the ground station communications in the private UAV network. Each Wireshark frame trace shows the successful communication between devices in this network. Ping command was also used with the IP address of each device to verify the communication from the ground station to each device connected in this network. Each ping command gave the successful reply messages without any packet loss that confirm the communication from the ground station to each device in the network. Furthermore, we used the traceroute command with the IP address of each WAP and smartphone camera. The output of traceroute command did not show any intermediate devices IP address, and it confirmed that there was no routing, but, rather, there was switching at the WAPs. As the spanning tree was enabled, the frames were switched from one WAP to another until they were received at the destination.

We successfully implemented the Storm 4 Mini private UAV network testbed and checked the communication between UAV and UAV to the ground station by conducting various testcases. However, there were some issues in this private UAV network testbed; first, we could only transmit the video signals from one UAV to another through the Wi-Fi frame transmission while each UAV was controlled with its own remote controller through analog signals. We received video data from each UAV to the single ground station, but this single ground station could not control all the UAVs due to their analog control signals. We did not fly these UAVs during the experimental testbed due to several reasons; first, they are controlled by remote controller, and we needed six expert persons to fly these UAVs by their controller and it was hard to manage; second, the flying of these UAVs was not stable and adding the extra payload (WAP, power bank and iPhone) made it worse to fly. It was still required to implement this private UAV network in which we could fly all the UAVs from a single ground station and receive videos from all UAVs to this ground station to test the communication between UAVs and UAV to the ground station. It was only possible with the smartphone operated UAVs that could be controlled by digital Wi-Fi signals. Accordingly, we conducted further research and found that UAV manufacturing companies designed smartphone operated UAVs which can also carry some weight, such as the AR 2.0, DJI and 3DR Solo models (the details of each of these models are presented in Appendices B, C and D) [8], [10], [11]. The issue was how to buy these new UAV models since we had fully allocated our research budget. In the later stage of the research, we got an opportunity to test this MAC layer communications between UAV and UAV to the ground station through another private UAV network testbed with AR 2.0 UAVs. These UAVs were bought for another project but were available to use later in this project. An AR 2.0 UAV is a smartphone operated UAV and uses digital signals to transmit the video to the smartphone application and also controlled by smartphone application through digital control signals. We implemented the MAC layer communication between UAVs and UAV to the ground station in the second private UAV testbed network with the flying of AR 2.0 UAVs. A detailed description of these communications in the second private UAV network is presented in the next section.

3.10 Experimental testbed of AR 2.0 UAV private UAV network for communication verification

A private UAV network with AR 2.0 UAVs is being developed by UAV network research group at CQUniversity Sydney campus with the team of my principal supervisor Rohan de Silva, myself and another research student Sandaruvan Rajasinghege. For this AR 2.0 private UAV network project, six UAVs were bought to implement a private UAV network.

We got the opportunity to use these UAVs to test the communication between UAVs and UAV to the ground station at the MAC layer by switching through UAV nodes with the flying of UAVs. The AR 2.0 UAV was designed by a company named Parrot with some unique features. We can simply download the free available smartphone application to control this UAV and receive the live video streaming from the UAV cameras through this application. The interesting part of the AR 2.0 UAV is that we can modify the code for control and video transmission according to our needs. Another advantage of the AR 2.0 UAV is its two built-in cameras and, therefore, we did not require a separate Wi-Fi camera for each UAV to test the communication for live video streaming form UAVs to a ground station in this private UAV network. However, we still used the additional WAP (mounted on top of each UAV) to communicate with other UAVs in this network. The main difference between in communication in this private UAV network with the Storm 4 Mini private UAV network is their control mechanism. In the private UAV network with the Storm 4 Mini UAVs, each UAV can only be controlled by its remote controller and video data is transmitted through UAVs by forming a network. Furthermore, these UAVs are not staying in a stable position during the flight, and it is hard to control their movement through analog signals when they are in flying mode. As such, we did not fly Storm 4 Mini UAVs during any experiments to test the communication between UAVs and UAV to the ground station. In contrast, Wi-Fi digital signals are used to control the AR 2.0 UAV from a smartphone application. This is an advanced UAV model, which can stay in a stable position during the flight—that is, it contains various sensors which help it to fly safely. Accordingly, some indoor flight testing with the AR 2.0 UAVs was performed to test the communication between UAVs and UAV to the ground station for this private UAV network. In this UAV network, we tested the communication between UAVs and UAV to the ground station by transmitting the control, feedback and video signals.

Before the implementation the communication at MAC layer between UAVs and UAV to the ground station in this AR 2.0 private UAV network, we first studied the communication protocols used between AR 2.0 UAV and its ground controller to gain a better understating of its communication system. The AR 2.0 UAV encapsulates its data (video and feedback) into a Wi-Fi frame and transmits to the ground controller (smartphone application). The ground controller which receives this Wi-Fi frame decapsulates it to retrieve this data. This process is also followed during the control data transmission from the ground controller to the UAV (control command transmission). These data encapsulation and decapsulation processes are accomplished through the set of layered protocols defined in the TCP/IP protocol stack. The

TCP/IP suite is a layered protocols stack, which is used to interconnect network devices in a network. This protocol stack is presented in Fig. 3.36.

Fig. 3.36 shows four layers in the TCP/IP protocol stack—namely, the Application Layer, Transport Layer, Internet Layer and Network Interface Layer. The application layer is responsible for allowing the access of network resources and having the application protocols for file transfer, email or remote login. More specifically, the transport layer defines the protocols for end-to-end data transmission. This layer is responsible for transferring the whole message, which is known as the user datagram, between two communication devices in the network. The Internet layer is used for communication between two devices in the network. This layer is responsible for sending the individual datagram between two communicating devices in the network. The network interface layer defines how the data will send through the physical medium from one device to another in the network [68].

Application Layer
Transport Layer
Internet Layer
Network Interface Layer

Fig. 3.36 TCP/IP protocol stack

We have first to understand the role of each layer with its communication protocols that are used during the communication between the AR 2.0 UAV and its ground controller. Fig. 3.37 illustrates the communication between a UAV and its ground controller in which a UAV transmits its data (that is feedback and video) to the ground controller and receives the control command from the ground controller. For this communication, the ground controller first establishes a wireless connection with the UAV and sends the control commands to the UAV in the Wi-Fi frame. The formation of this Wi-Fi frame is accomplished by passing the data to the layer below in this protocol stack. Each layer adds its header to the incoming data and passes it to the next layer (see Fig. 3.37). The Network Interface layer transmits this frame over the wireless physical medium to the UAV. The UAV receives this Wi-Fi frame and retrieves its data by removing the header at each layer, as shown in Fig. 3.37.

To get the details about the communication protocols used by AR 2.0 UAV, we simply set up our notebook computer as a ground station and installed a free application to control this UAV. We connected this notebook computer with the AR 2.0 Wi-Fi network and sent the take-off commands through the application to the UAV. This control command allows UAVs to fly at one-meter height and stay there in a stable condition until it gets the next command from the application. At the same time, the application receives live video streaming from the front camera of the UAV. We used the Wireshark and captured the Wi-Fi frame that was transmitted between the UAV and the notebook computer. Since we knew the IP addresses of the AR 2.0 UAV and the notebook computer, we easily identified what type of frames were going out and coming in from and to the ground station. We examined these frames and got the information about the communication protocols used by this UAV.

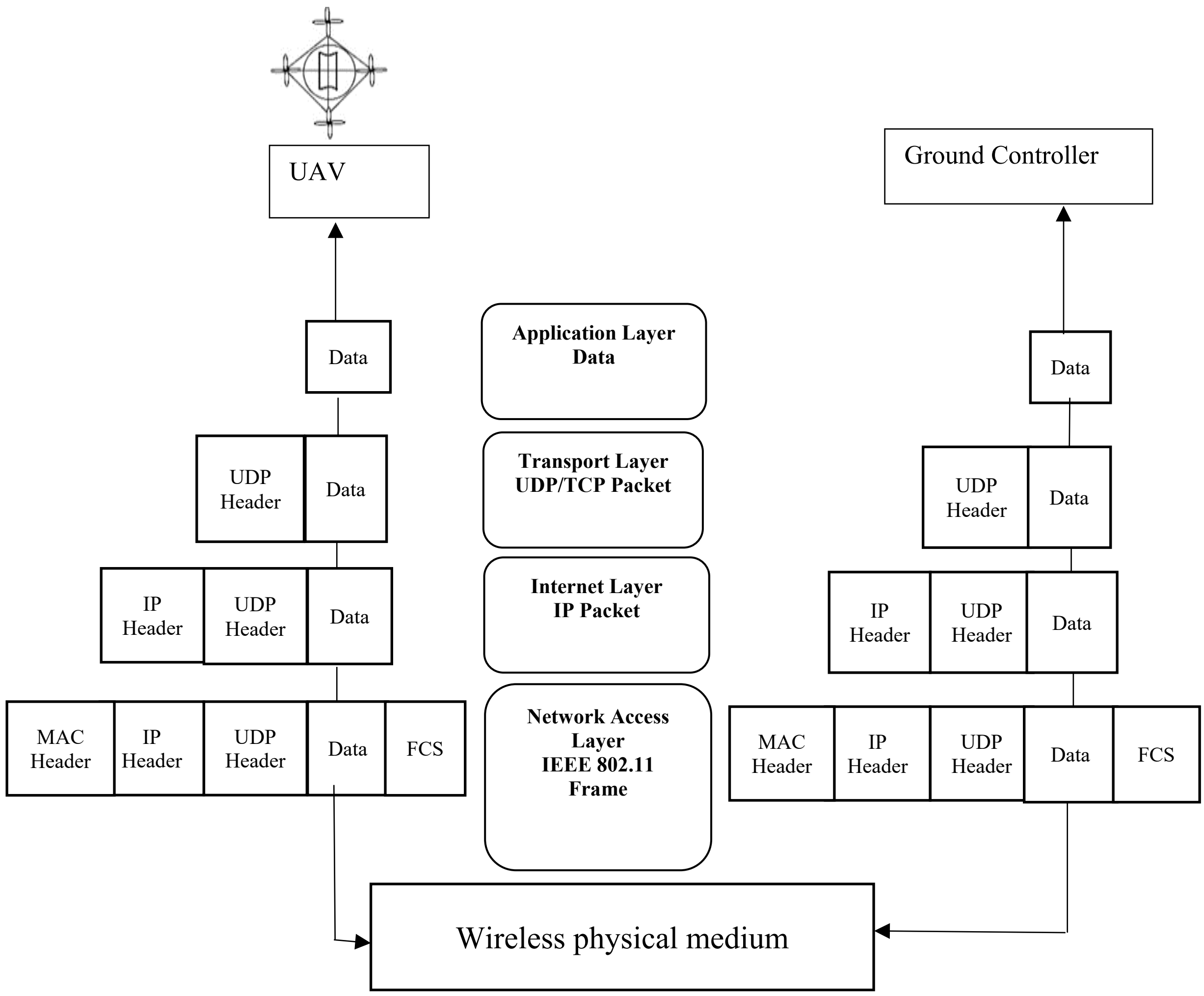


Fig. 3.37 Communication between UAV and the ground controller

In the AR 2.0 UAV and the ground station communication, the AR 2.0 UAV supports the Wi-Fi IEEE 802.11 b/g/n protocols to communicate with other wireless devices [10]. The two IP protocols predominantly used by the TCP/IP model at the Internet layer are Internet Protocol Version 4 (IPv4) and Version 6 (IPv6). As discussed earlier, we captured the Wi-Fi frame in the Wireshark during the communication between UAV and the ground station for AR 2.0 UAV. We studied three different Wi-Fi frames captured in the Wireshark, that is, control, feedback and video frames, to examine the IP protocols for the communication between UAV and the ground station. We found that all the three frames have the IPv4 as an IP protocol. The transport layer has two main protocols named Transmission Control Protocol (TCP) and User Datagram Protocols (UDP) used for communication between two network devices. The AR 2.0 UAV uses UDP protocol for control and navigation data, and TCP protocol is used for video data transmission between UAV and ground station, and it was confirmed with the Wireshark captured Wi-Fi frames. This UAV uses the File Transfer

Protocol (FTP) and Telnet application layer protocols to transfer files to a controller, as well as to create a remote UAV login. Fig. 3.38 summarises these protocols used for communication between the AR 2.0 and the ground controller, with reference to the AR 2.0 private UAV network.

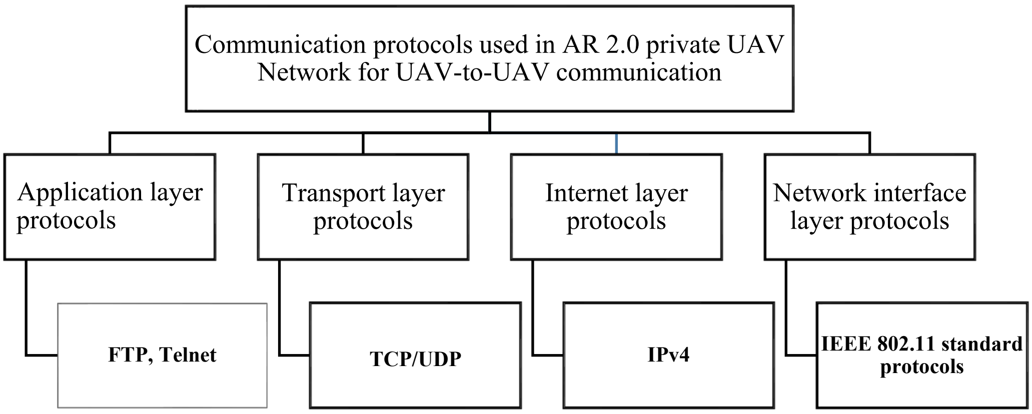


Fig. 3.38 Communication protocols used in AR 2.0 private UAV network

Once, we studied the communication protocols that are used by AR 2.0 UAV for communication with the ground controller; then we implemented the communication between UAVs and UAV to the ground station at MAC layer by switching through UAV node for the flying AR 2.0 private UAV network.

3.10.1 Hardware components details

Most of the hardware components in the AR 2.0 UAV network experimental setup were the same as the Storm 4 Mini private UAV network. For this network, we used the same WAP model, power bank and notebook laptop. We did not require any Wi-Fi camera for this network, as the AR 2.0 model contains two built-in cameras. We tested the communication in this network with three AR 2.0 UAVs, three Edimax WAPs, three power banks and a ground station notebook laptop. As the hardware components for the AR 2.0 UAV's network are similar to those for the Storm 4 Mini UAV network (discussed in the previous section), only a discussion of the AR 2.0 UAV is presented in the next subsection.

3.9.1.1 AR 2.0 UAV model

We provided a brief overview of this UAV earlier with its communication protocols. This UAV does not require a separate controller to fly, nor to enable live video streaming from it. The manufacturer of this UAV, Parrot, currently provides a free app for flying and live video streaming. This app can be installed on any smartphone or tablet to control this UAV as well as to watch the live video streaming captured by this UAV. Parrot provides some common settings to support connectivity across all of their AR 2.0 UAVs. First, we powered up the UAV and then we set up our mobile phone Wi-Fi connection with the AR 2.0 UAV. Once our smartphone is connected with the UAV, we then launch the free flight app and, in the pilot mode of this app, we commence flying the UAV. This app also contains built-in live streaming capabilities. We can watch the live video streaming from the cameras of the UAV while this UAV was at the ground level or in flying mode. The default IP address of this UAV is 192.168.1.1. Additionally, it can fly for 8–12 minutes and capable of carrying some weight that is required to form this private UAV network (a detailed description of the AR 2.0 UAV is given in Appendix B) [10].

3.10.2 Device configuration

To implement the communication in this AR 2.0 private UAV network, we configured each WAP as well as the AR 2.0 UAV. The steps to configure. the WAP devices for this network were the same as the private network using the Storm 4 Mini UAVs. However, we changed the IP address and SSID of each WAP in this network to verify the communication path between the UAVs easily. We did not use any external Wi-Fi camera in this network; however, we configured each AR 2.0 UAV to make it as a wireless client for its corresponding WAP. The other issue in this testbed network was a selection of the frequency band for the WAPs since the AR 2.0 does not support on the 5-GHz frequency band. Therefore, all the WAPs were configured over the 2.4-GHz frequency band in this private UAV testbed network.

3.9.2.1 WAP configuration

For each WAP in this network, we assigned a different IP address as well as a different SSID, but we kept the common password to access all of these WAPs. All of the WAPs were configured over the 2.4-GHz band for this network, and we assigned a common channel (that is, Channel 6) to each WAP, as this channel usually contains less traffic compared with other channels. Table 3.7 presents the details of the configuration of each WAP for the AR 2.0 private UAV network testbed.

Table 3.7 WAPs configuration for the AR 2.0 private UAV network

IP addresses	10.10.10.21 to 10.10.10.23
Wireless	2.4-GHz network
SSID	net1 to net3
Security	WPA2
Password	Testnetwork
Mode	AP Bridge WDS
Common channel	6
MAC address	Each UAV had the MAC address of the previous and next UAV
Wireless spanning tree	Enabled
Transmitting power	15 per cent

3.9.2.2 AR 2.0 UAV configuration

In this private UAV testbed network, we configured each AR 2.0 UAV to make them into wireless clients for their corresponding WAP. With the help of the smartphone app, we changed the SSID of each AR 2.0 UAV and then assigned a new SSID (that is, from Dnet1 to Dnet3) to the three AR 2.0 UAVs to identify each UAV in this private UAV network easily. The AR 2.0 provides a license-free Software Development Kit (SDK), and with the help of some basic commands, we configured each AR 2.0 UAV. We followed some steps that are mentioned below to change the configuration for each AR 2.0 UAV in this private UAV network.

Step 1. Once we powered on the AR 2.0 UAV, our laptop was able to detect the SSID of this UAV. We connected our laptop to the AR 2.0 Wi-Fi network, as no password was required to connect.

Step 2. We changed the IP address of the AR 2.0 UAVs and changed them to the wireless clients for their corresponding WAPs with the help of some basic commands.

Table 3.8 Configuration of AR 2.0 UAVs for the private UAV network testbed

AR 2.0 UAVs SSID	**IP addresses**	**Corresponding WAPs**
Dnet1 to Dnet3	10.10.10.11 to 10.10.10.13	10.10.10.21 to 10.10.10.23

Table 3.8 shows the configuration of each AR 2.0 UAV for the experimental testbed. After performing the above two steps, we verified the connections between the UAV and its corresponding WAP to ensure that AR 2.0 UAV became the wireless client of the WAP. We connected our laptop with the net1 network and pinged 10.10.10.21 address and received successful reply messages. We then used the 'ping' command to ping the UAV with the IP address 10.10.10.11 (which was the new IP address of the UAV) and received successful reply messages from the UAV confirming the UAV was working as a wireless client to the WAP. For each AR 2.0 UAV, we performed the same two steps to make them as a wireless client for their corresponding WAP in this private UAV testbed network and also verified the communication with the ground station with ping commands. Although AR 2.0 UAVs are designed to operate with the help of a smartphone app, we used this UAV to form a private network controlled by a single ground station (that is, notebook laptop). Therefore, we downloaded an additional library on the laptop in order to control the UAV from a notebook laptop.

3.9.2.3 Configuration of the ground station

We assigned a static IP address (that is, 10.10.10.50) to the laptop to enable it to work as a ground station in this network. Furthermore, the laptop was always connected to the 'net1' WAP, which furthered communicated with other WAPs in the network.

3.11 Communication verification in the AR 2.0 private UAV network

We implemented a second practical testbed with flying AR 2.0 UAVs to form a private UAV network and performed various tests to verify the communication between UAVs and UAV to the ground station. In the following section, we discuss a test case that was performed to test the communication between UAVs and UAV to the ground station for the flying AR 2.0 UAVs private network practical testbed.

3.11.1 Private UAV network with flying UAVs

We configured each UAV by changing their IP address and made it a wireless client of its corresponding WAP. We connected all three UAVs in tandem and placed each WAP near the corresponding UAV. We connected our laptop (that is, the ground station) to WAP 1 network through the SSID net1. The laptop was communicating directly with net1 and this WAP 1 was communicating with WAP 2 of net2 and WAP 2 was communicating with WAP 3 of net3 through switching in MAC layer. The IP addresses of the three WAPs were from 10.10.10.21 to 10.10.10.23, with their corresponding UAV clients having IP addresses from 10.10.10.11 to 10.10.10.13. To test the connectivity, we pinged all the devices one by one from the ground station and received successful reply messages from all of them. We sent the control commands for each UAV from the laptop to take-of and hover at one-meter height. The experimental setup for this private UAV network while UAVs hovering is shown in Fig. 3.39.

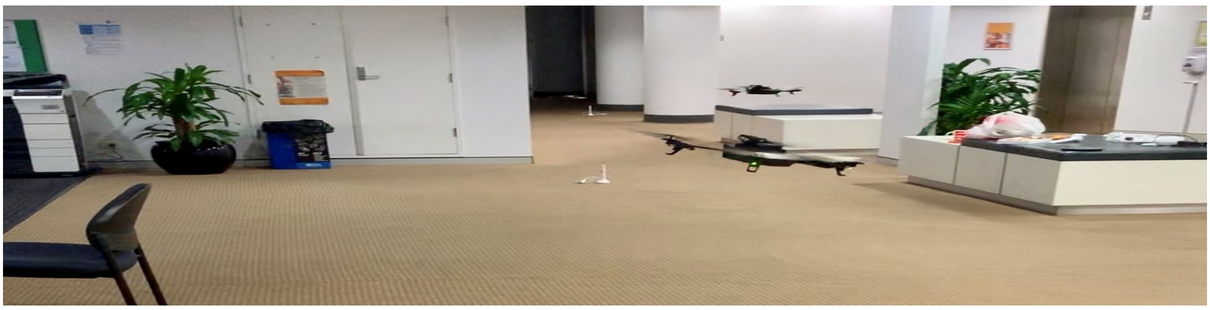
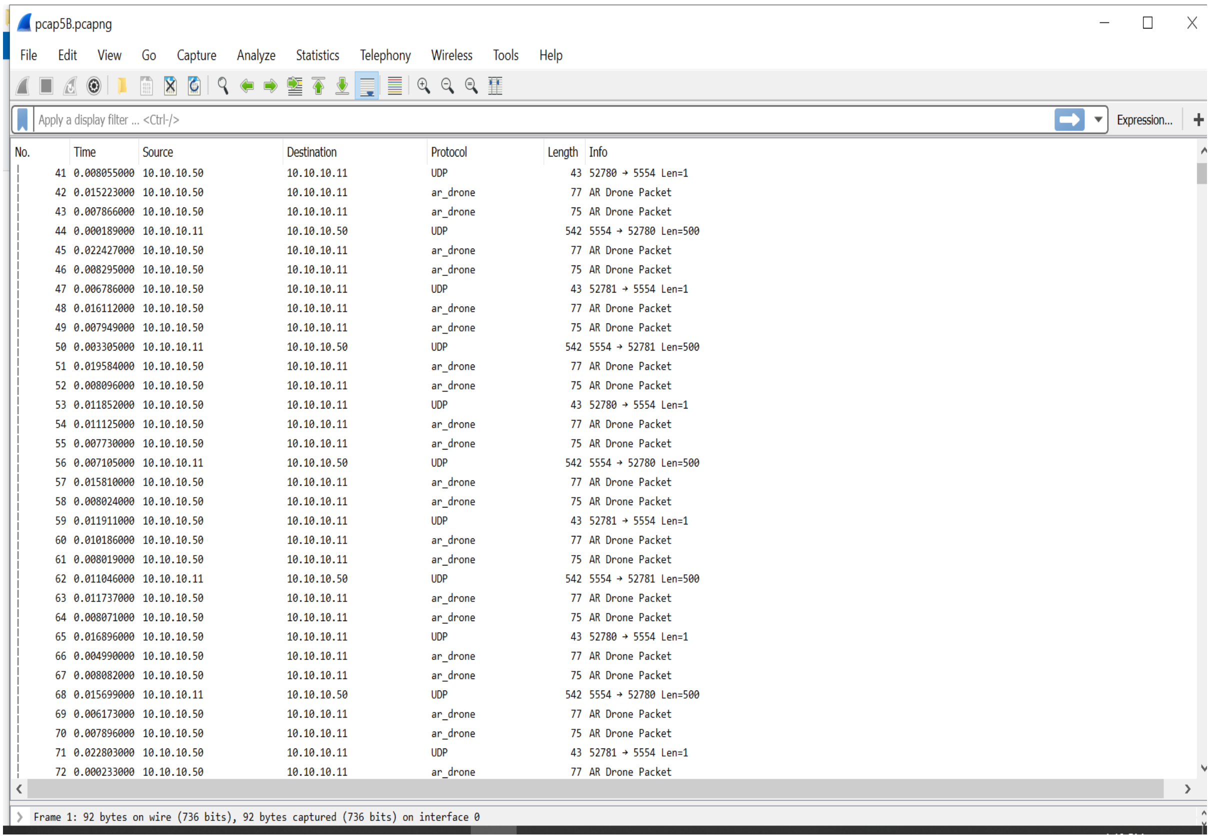


Fig. 3.39 Experimental setup for flying AR 2.0 private UAV network

Disclaimer: In these experiments, we flew UAVs at one meter high and such flights do not require CASA approval [69]. These experiments were conducted on Sundays at CQUniversity Sydney campus in the presence of my principal supervisor under the safe and controlled conditions. Level 5 of the CQUniversity Sydney campus was booked for these experiments so that the other students could not come there. As such, there was no risk to any CQU staff or student from flying UAVs indoors in these experiments.

We undertook extensive testing to verify the end to end communication in this flying AR 2.0 private UAV network and achieved successful results for both UAV-to-UAV and UAV to the ground station communication. First, we verified the end to end communication in this flying AR 2.0 UAV network by switching off the intermediate WAP 2 in this network. Then, the UAV 3 that was at the far end in this network started moving abruptly as it was unable to receive the control commands from the ground station. That shows the end to end communication in this flying AR 2.0 UAV network happens through WAPs that were configured for the communication at the MAC layer between UAVs. Second, with WAP 2 back on the network, we captured the Wireshark frames to verify the communication between UAV 1 and the laptop. The captured Wireshark frames during the flying of UAVs are shown in Fig. 3.40.



No.	Time	Source	Destination	Protocol	Length	Info
41	0.008055000	10.10.10.50	10.10.10.11	UDP	43	52780 → 5554 Len=1
42	0.015223000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet
43	0.007866000	10.10.10.50	10.10.10.11	ar_drone	75	AR Drone Packet
44	0.000189000	10.10.10.11	10.10.10.50	UDP	542	5554 → 52780 Len=500
45	0.022427000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet
46	0.008295000	10.10.10.50	10.10.10.11	ar_drone	75	AR Drone Packet
47	0.006786000	10.10.10.50	10.10.10.11	UDP	43	52781 → 5554 Len=1
48	0.016112000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet
49	0.007949000	10.10.10.50	10.10.10.11	ar_drone	75	AR Drone Packet
50	0.003305000	10.10.10.11	10.10.10.50	UDP	542	5554 → 52781 Len=500
51	0.019584000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet
52	0.008096000	10.10.10.50	10.10.10.11	ar_drone	75	AR Drone Packet
53	0.011852000	10.10.10.50	10.10.10.11	UDP	43	52780 → 5554 Len=1
54	0.011125000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet
55	0.007730000	10.10.10.50	10.10.10.11	ar_drone	75	AR Drone Packet
56	0.007105000	10.10.10.11	10.10.10.50	UDP	542	5554 → 52780 Len=500
57	0.015810000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet
58	0.008024000	10.10.10.50	10.10.10.11	ar_drone	75	AR Drone Packet
59	0.011911000	10.10.10.50	10.10.10.11	UDP	43	52781 → 5554 Len=1
60	0.010186000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet
61	0.008019000	10.10.10.50	10.10.10.11	ar_drone	75	AR Drone Packet
62	0.011046000	10.10.10.11	10.10.10.50	UDP	542	5554 → 52781 Len=500
63	0.011737000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet
64	0.008071000	10.10.10.50	10.10.10.11	ar_drone	75	AR Drone Packet
65	0.016896000	10.10.10.50	10.10.10.11	UDP	43	52780 → 5554 Len=1
66	0.004990000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet
67	0.008082000	10.10.10.50	10.10.10.11	ar_drone	75	AR Drone Packet
68	0.015699000	10.10.10.11	10.10.10.50	UDP	542	5554 → 52780 Len=500
69	0.006173000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet
70	0.007896000	10.10.10.50	10.10.10.11	ar_drone	75	AR Drone Packet
71	0.022803000	10.10.10.50	10.10.10.11	UDP	43	52781 → 5554 Len=1
72	0.000233000	10.10.10.50	10.10.10.11	ar_drone	77	AR Drone Packet

Frame 1: 92 bytes on wire (736 bits), 92 bytes captured (736 bits) on interface 0

Fig. 3.40 Wireshark captured frame to verify the communication in the AR 2.0 network

We can see from Fig. 3.40, two types of packets, ar _drone and UDP are transmitted during the communication between UAV 1 and the laptop. For example, packet No. 49 is an ar _drone control packet sent from the IP address 10.10.10.50 (laptop) to IP address 10.10.10.11 (UAV1). In contrast, packet No. 50 is an ar _drone feedback packet received from the IP address 10.10.10.11 (UAV1) at the laptop. This shows that end to end communication happens in the AR 2.0 private UAV network at MAC layer while UAVs were flying as well.

3.12 Chapter summary

In this chapter, we discussed the two private UAV testbed networks that were used to verify the communication at the MAC layer. We first verified the communication between UAVs and UAV to the ground station for this network through the OPNET simulator and then implemented these communications in the private UAV testbed networks. We also outlined the practical problems faced during the implementation of these communications in this chapter. We undertook various test cases to ensure the communication between UAVs and UAV to the ground station through these UAV testbed networks, and it is also covered in this chapter.

The main aim of these practical testbed networks was the fast transmission of data between UAVs and UAV to ground station by implementing the communication at the MAC layer by switching through UAV nodes. We encountered several problems during the practical implementation of the private UAV network. The first problem was the communication interference within the Storm 4 Mini UAV network. In our early experiments with this network, we used the 2.4 GHz network to transfer the video; however, once we started the UAVs, which also worked at the same frequency band, the video data froze, and we were unable to receive the live video stream. We found a solution to this issue and sent the video data via the 5 GHz frequency band, and control signals were operated on the 2.4 GHz frequency. In both private UAV networks, we encountered the problem of video delays from each UAV to the ground station due to the use of a single common communication channel. We were unable to change this communication channel due to the WDS bridge mode configuration of the WAP and these communication channels were standardised. Therefore, a single channel was used to transmit the control, feedback and video signals between UAVs and that introduced a delay at each UAV in this network.

Furthermore, the experimental testbed presented in this study will help the UAV manufacturing companies become more aware of the practical issues which may arise in this network setup—as well as the strategies to address them—when designing UAVs which can communicate with each other.

Chapter 4

Control and data channels for the private UAV network

4.1 Introduction

In Chapter 3, we discussed the implementation of practical testbed networks to verify the communication between UAVs and UAV to the ground station in the private UAV network. We could not resolve some issues in these experimental testbeds due to the preexisting design of UAV models (Storm 4 Mini and AR 2.0) and it was not possible to change the hardware of these UAV models. One of these unresolved issues was the live video streaming delay at the ground station. In the experimental testbed network, the ground station received the live video streaming from each UAV with some delays. The cause of these video delays was the use of a common single communication channel between UAVs for the communication in those experimental testbed networks. Furthermore, this single channel was shared for the transmission of control, feedback and video signals between UAVs. Therefore, each UAV processor had to do some time-sharing to process these signals, and it introduced a video processing delay at each UAV. This delay was greater for the far end UAV node in this testbed network. This is because the video captured by the far end UAV node needed to pass through each of the intermediate UAVs before it reached to the ground station. We could not resolve this issue due to the standardized communication channels used by two frequency bands (2.4 and 5 GHz) in the experiment testbed network. However, this requires the creation of different channels for data and control information.

As such, in this chapter, we introduce the new channel allocation to transfer user data and control signals in private UAV network operating in 2.4 GHz as well as 5 GHz frequency bands. We created the channels considering the minimum bandwidth required for video data transmission by the cameras mounted on commercial UAVs and the bandwidth required for control data transmission.

The remainder of this chapter is as follows. In Section 4.2, the requirements for these new channels are discussed, and the frequency spectrum used for UAV to the ground station communication for the current commercial UAVs are examined. Section 4.3 of this chapter discusses the existing IEEE 802.11 Wi-Fi standards and their channel classification. The channel allocation of the 2.4 and 5 GHz frequency spectrum is also covered in this section.

Accordingly, Section 4.4 outlines the bandwidth requirements for video and control transmission and their use in the development of the new control and video channels for the private UAV network. The creation of new control and data channels for private UAV network are explained in Section 4.5. Additionally, Section 4.6 describes the channel bonding used for high-quality video transmission for this UAV network. Finally, a chapter summary is provided in Section 4.7.

4.2 Communication channels for the private UAV network

In Chapter 3, we discussed the communication mechanism between UAVs and UAV to the ground station through two private UAV testbeds network in which UAVs were connected in tandem. In both practical testbeds, the video data was transmitted through the Wi-Fi signals from each UAV to the ground station. As such, a single ground station received the live video streaming from the six UAV nodes in this network. As discussed in Chapter 3, the ground station received these live video streaming with some delay from each UAV in the private UAV testbed networks. We conducted a simple experiment to get a better understanding of these video delays. We formed a simple network with three WAPs and three iPhones that act as Wi-Fi cameras for their corresponding WAPs. We mounted these three iPhones to their corresponding WAPs. These WAPs were configured in WDS bridge mode same as the experimental testbed network and we manually gave the IP address to each iPhone in the same subnet and installed a free Wi-Fi camera app on it. We then positioned the WAPs next to each other in such a way that the first and second WAP could only communicate with the help of the intermediate WAP (that is, WAP 2). We connected the laptop to the first WAP (that is, WAP 1) wireless network and accessed three live video streams at the laptop from the iPhone cameras connected with the WAP. In this communication, the laptop could only access the videos from the iPhone's camera through their WAPs. We observed the delays in live videos that were received at the ground station. In the actual private UAV network testbed that was discussed in Chapter 3, each of these WAPs needed to communicate with the UAV itself (that is, AR 2.0 private UAV network) before they transmitted the control, feedback and data signals to other UAV in the network; therefore, greater video delays were introduced by each UAV and, furthermore, the ground station experienced greater delays in receiving the video signals from each UAV.

We investigated about the video transmission delay problem and found that these video delays were introduced in the network due to two basic reasons: first, we used a common communication channel at each UAV, and second was the sharing of this single channel for data (control, feedback and video) transmission between UAVs. Furthermore, these wireless communication channels were developed and standardised on different frequency bands (2.4 GHz and 5 GHz). The private UAV network implemented in our practical testbed, used 5 GHz for video transmission as the control signals were operated on the 2.4 GHz frequency with remote controller (that is, Storm 4 Mini private UAV network). A single frequency band (2.4 GHz) was used to implement the AR 2.0 private UAV network since video, control and feedback signals transmitted through each UAV itself and we could not change the frequency band for this UAV model. It should be noted here, the live video streaming delay at the ground station for both UAV networks was different. That is, in the Storm 4 Mini private UAV network, each UAV WAP processor had to process only the video signals as the control singles were transmitted through the remote controller to each UAV. Whereas, in the AR 2.0 private UAV network, all the three signals (control, feedback and video) were processed by a UAV WAP processor. Henceforth, we are considering only the AR 2.0 private UAV network here to better understand this video delay problem as all three signals were transmitted between UAVs while they were communicating with each other. Each UAV in the AR 2.0 private UAV network shares this single channel to transmit control, feedback and video signals to other UAVs as well as to the ground station. Furthermore, each channel in the frequency band (that is, 2.4 or 5 GHz) has a limited capacity; therefore, UAV WAP processor had to perform some time-sharing to process these signals and the ground station experienced a delay in receiving video signals from each of the UAVs in the network.

In the AR 2.0 private UAV network, we configured each UAV to operate on the same frequency channel and allowed them to share control and user data transmitted between two UAVs and UAV to the ground station. However, this required the creation of new channelisation in the private UAV network to overcome the signal transmission delays from one UAV to other and UAV to the ground station. It should be emphasised here that these are not frequency channels; rather, they are data channels whose capacities are expressed in bits per second (bps). For this purpose, we examined the existing IEEE 802.11 Wi-Fi standards and their corresponding communication channels and created new channels by considering minimum bandwidth requirement for transmission of video and control data between UAVs.

4.2.1 Demonstration of unequal bandwidth sharing of the WAP

It was noticed that control and feedback packets get delayed in the presence of video traffic when passing through the WAP. The packets were captured in Wireshark at the laptop computer without and with the WAP between AR 2.0 and the laptop computer.

Case 1: One UAV directly connected with the ground station without WAP

In this communication setup, a single UAV was directly connected with the laptop computer as shown in Fig. 4.1 and the UAV streamed video continuously to the ground station. Wireshark traces for feedback and control packets were captured and the graphs for the arrival time point of them were found and shown in Fig. 4.2 and Fig. 4.3 respectively.

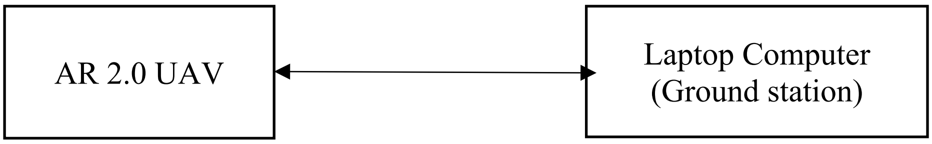


Fig. 4.1. UAV to the ground station communication without WAP

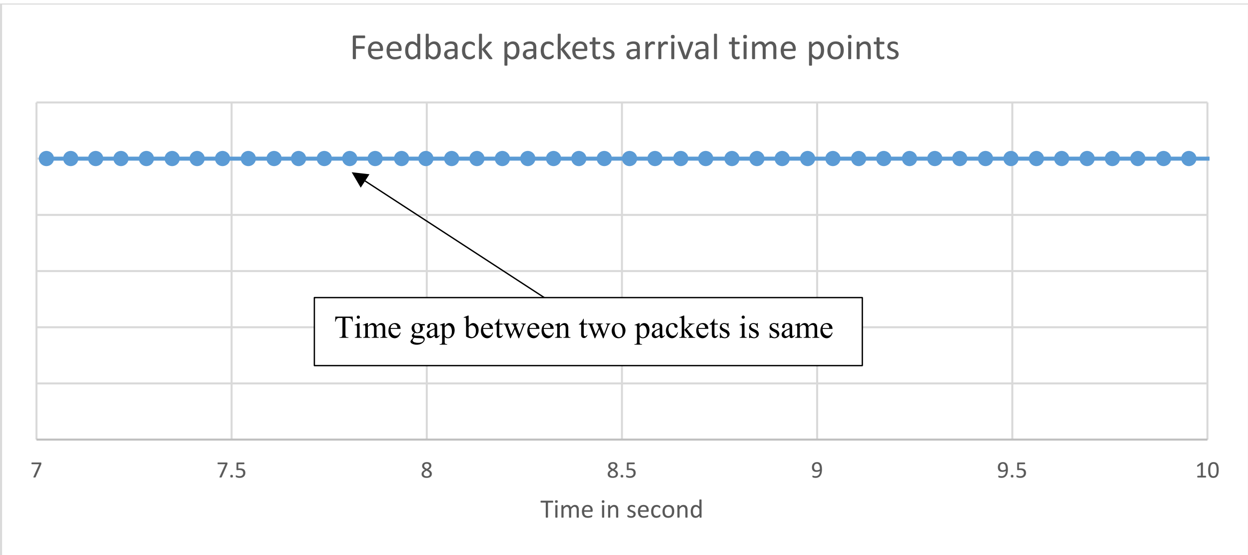


Fig. 4.2 Feedback packets arrival time points for the single UAV with video data without WAP

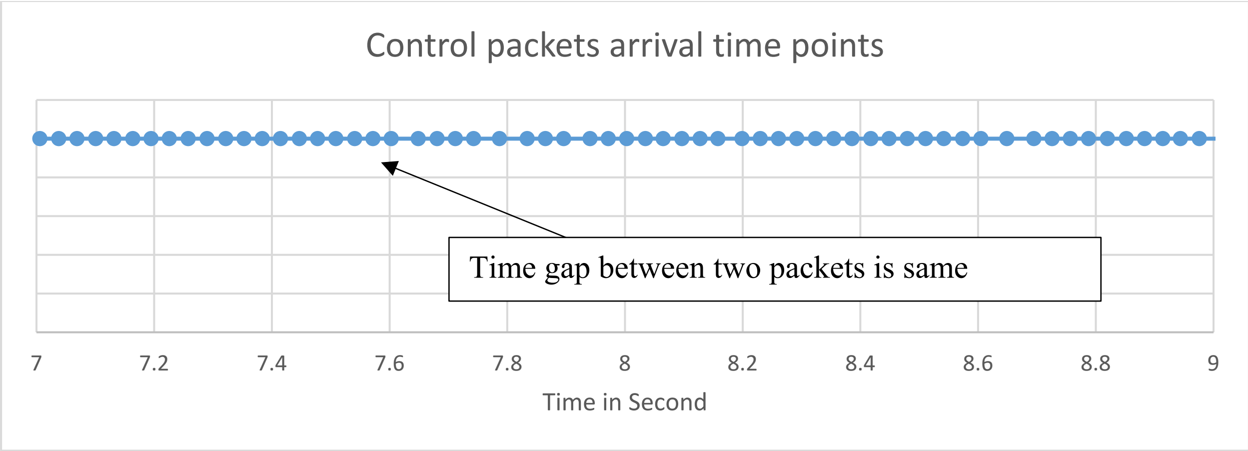


Fig. 4.3 Control packets arrival time points for single UAV with video data without WAP

As shown in Fig. 4.2 and Fig. 4.3, the time gap between two packets is the same.

Case 2: One UAV connected through the WAP with the ground station

A single UAV was connected through the WAP with the laptop computer as shown in Fig. 4.4 and the UAV streamed video continuously to the ground station via the WAP. Wireshark traces for feedback and control were captured and the graphs for the arrival time point of them were found and shown in Fig. 4.5 and Fig. 4.6 respectively.

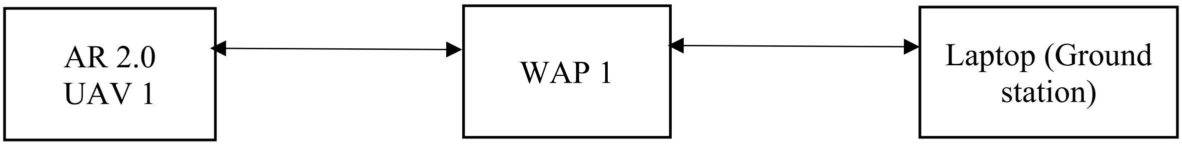


Fig. 4.4 UAV to the ground station communication with WAP

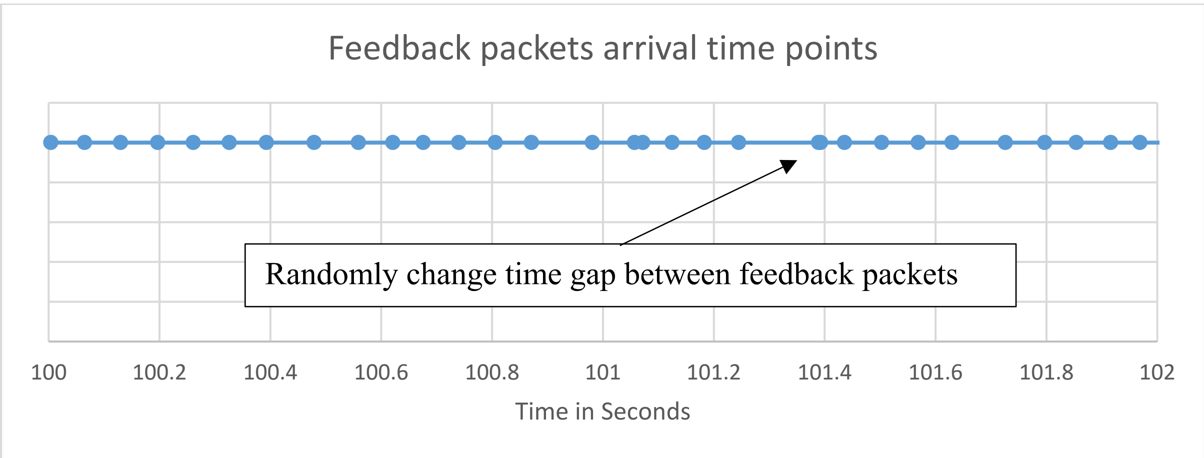


Fig. 4.5 Feedback packets arrival time points for the single UAV with video data with WAP

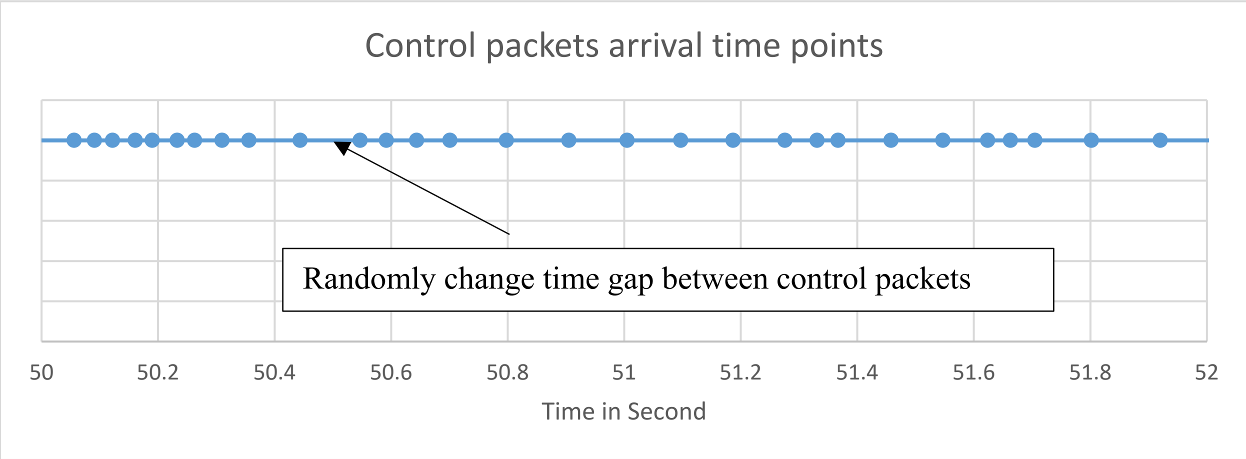


Fig. 4.6 Control packets arrival time points for the single UAV with video data with WAP

It can be clearly seen in Fig. 4.6 and Fig. 4.7 that the time gaps between packets have randomly changed. This is due to the buffering and processing delay at the WAP. This confirms that a WAP does not equally share the bandwidth for all traffic and channelisation is required.

4.3 IEEE 802.11 Standards and channel classifications

The IEEE 802.11 published various standards for communication and data transmission between wireless devices. Private UAV networks are also a type of wireless network and, therefore, these existing IEEE 802.11 standards are used to transmit control, feedback and video data between UAVs and UAV to the ground station. The commercial UAVs that are currently available in the market which operate via Wi-Fi used these IEEE 802.11 standards for UAV to the ground station communication. In the practical testbed network, WAPs were used to enable communication between two UAVs. These WAPs also used these IEEE 802.11 standards to communicate with other wireless devices. In future, if a new UAV model will be designed with in-built capabilities to communicate with other UAV, these IEEE 802.11 standards for communication will also be used in it. In this chapter, we first discuss the existing IEEE 802.11 standards to determine their maximum data transmission rates, frequency bands and communication channel bandwidths and then new channelisation for the private UAV network is presented.

4.3.1 Summary of existing IEEE 802.11 standards

The IEEE 802.11 wireless communication standards were developed in the past, and the enhancement in these standards is continuously going on. In this section, the history of the IEEE 802.11 standards is examined to better understand their current status. Legacy 802.11 was the first IEEE 802.11 family standard, which was published in 1997. The data rates associated with this standard were 1 Mbps and 2 Mbps, and it uses a 2.4-GHz spectrum for data transmission [70]. In 1999, IEEE 802.11a was published, which supports 6, 9, 12, 18, 24, 36, 48 and 54 Mbps data rates. This increase in the rates was due to the different modulation techniques used by this standard. IEEE 802.11a operates on the 5 GHz frequency spectrum for data transmission, and it has 12 non-overlapping communication channels [71]. In the same year, IEEE published one more standard (that is, 802.11b), that works over the 2.4 GHz frequency spectrum. The maximum data transmission rate of this standard is 11 Mbps [72]. In 2003 IEEE 802.11g was released, that can support the maximum data rate of 54 Mbps and operated over the 2.4 GHz frequency spectrum. Additionally, it contains three non-overlapping channels for data transmission [73]. In 2009, IEEE 802.11n was released, and it operates on both the 2.4 and 5 GHz frequency spectrums. The maximum data transmission rate of this standard is 54 Mbps; however, when it operates at 40 MHz with three transmitters and receivers, the maximum data rate increases to 450 Mbps. This theoretical data rate in this standard is achieved with single-user multiple input and multiple output (MIMO), which employs three spatial streams with 64 Quadrature Amplitude Modulation (QAM) modulation, as shown in Table 4.1 [74].

Table 4.1 Maximum data transmission rates for IEEE 802.11n

Mode	20 MHz	40 MHz	20 MHz	40 MHz	20 MHz	40 MHz
Transmitter & receiver	1TX, 1RX	1TX, 1RX	2TX, 2RX	2TX,2RX	3TX,3RX	3TX,3RX
Maximum rate	72.2 Mbps	150 Mbps	144.4 Mbps	300 Mbps	216.7 Mbps	450 Mbps

IEEE 802.11ac WAVE 1 was released in 2013, with a maximum data rate of 866 Mbps [75]. In 2016, the next version of IEEE 802.11 ac WAVE 2 was published and can support a data rate of up to 1.73 Gbps as it operates on the 160 MHz width channel for communication [76]. The theoretical speed of IEEE 802.11 ac can be achieved through the multiple-user MIMO and two spatial streams with 256 QAM modulation. Both of these IEEE 802.11ac standards work on a 5 GHz frequency spectrum for communication. The various data rates for these standards are presented in Tables 4.2 and 4.3 (below).

Table 4.2 Maximum data transmission rates for IEEE 802.11ac WAVE 1

Mode	40 MHz	40 MHz	80 MHz	80 MHz
Transmitter & receiver	1TX, 1RX	2TX, 2RX	1TX, 1RX	2TX,2RX
Maximum rate	200 Mbps	400 Mbps	433 Mbps	866.7 Mbps

Table 4.3 Maximum data transmission rates for IEEE 802.11ac WAVE 2

Mode	40 MHz	40 MHz	80 MHz	80 MHz	160 MHz	160 MHz
Transmitter & receiver	1TX, 1RX	2TX, 2RX	1TX, 1RX	2TX,2RX	1TX,1RX	2TX,2RX
Maximum rate	200 Mbps	400 Mbps	433 Mbps	866 Mbps	866 Mbps	1.73 Gbps

Table 4.4 Summary of IEEE 802.11 standards

Year	Standard	Frequency bands	Channel width	Max. the theoretical transmission rate in a single stream
1997	IEEE 802.11	2.4 GHz	22 MHz	1, 2 Mbps
1999	IEEE 802.11a	5 GHz	20 MHz	6, 9, 12, 18, 24, 36, 48, 54 Mbps
1999	IEEE 802.11b	2.4 GHz	22 MHz	1, 2, 5, 5, 11 Mbps
2003	IEEE 802.11g	2.4 GHz	20 MHz	6, 9, 12, 18, 24, 36, 48, 54 Mbps
2009	IEEE 802.11n	2.4 or 5 GHz	20/40 MHz	Up to 450 Mbps
2013	IEEE 802.11ac WAVE 1	5 GHz	20/40/80/160 MHz	Up to 866.7 Mbps
2016	IEEE 802.11ac WAVE 2	5 GHz	20/40/80/160 MHz	Up to 1.73 Gbps

Table 4.4 also outlines that the maximum speed of the 2.4 band is 450 Mbps and, furthermore, it adheres to the IEEE 802.11 n standard. On the other hand, the 5 GHz band has a maximum speed of 1.73 Gbps and adheres to IEEE 802.11ac WAVE 2. In this chapter, we discuss the development of new data and control channels for UAV to UAV and UAV to the ground station communication for the private UAV network with these two frequencies (that is, 2.4 GHz at 450 Mbps and 5 GHz at 1.73 Gbps). In the future, as new standards continue to become available on the market, new channels can be created via these standards for private UAV networks using the same methodology outlined in this study.

4.3.2 Channel allocation on the 2.4 and 5 GHz frequency spectrum

Most of the wireless devices communicate over the 2.4 and 5 GHz frequencies. Both of these frequencies have a pre-defined set of channels. Accordingly, when a wireless device communicates with other wireless devices over these frequencies, it uses one of the pre-defined channels from these respective frequencies as we saw in the practical testbed network in Chapter 3. The 2.4 GHz frequency spectrum is crowded compared with the 5 GHz frequency spectrum, and most Wi-Fi devices currently available use this frequency for communication. In a wireless communication between two devices through WAP, we can be configured this WAP manually to set a particular channel for the communication. This was the case in the private UAV testbed networks, in which we manually set the common channel of 36 to each WAP in the 5 GHz band setting and 6 to each WAP when we used the 2.4 GHz (that is, the Storm 4 Mini and AR 2.0 private UAV networks, respectively).

Table 4.5 Channels on the 2.4-GHz spectrum and their frequencies

Channel No.	**Frequency (MHz)**	**North America (FCC)**	**Japan**	**Europe (ETSI)**
1-11	2412-2462	Y	Y	Y
12	2467	N	Y	Y
13	2472	N	Y	Y
14	2484	N	802.11 b only	N

Table 4.6 Channels on the 5-GHz spectrum and their frequencies

Channel Number	**Frequency (MHz)**	**North America (FCC)**	**Japan**	**Europe (ETSI)**
36-48	5180-5240	Y	Y	Indoors
52-116	5260-5580	DFS	DFS / TPC	Indoors / DFS / TPC
120-128	5600-5640	N	DFS / TPC	DFS / TPC
132-140	5660-5700	DFS	DFS / TPC	DFS / TPC
149-165	5745-5825	Y	N	SRD

The total number of channels on the 2.4 GHz frequency spectrum is 14, and out of these, some are reserved and not available for users. Similarly, the 5GHz frequency spectrum also has pre-defined frequency channels. The summaries of these channel allocations in both spectrums are given in Tables 4.5 and 4.6. Therein, the following acronyms have been used. DFS, TPC, and SRD are used for Dynamic Frequency Selection, Transmit Power Control, and Short Range Devices, respectively. Additionally, the abbreviation ETSI stands for the European Telecommunication Standard Institute.

4.4 The bandwidth requirement for control and data channels

A private UAV network implemented in this study uses a single frequency channel from the 2.4 or 5 GHz frequency spectrum for UAV-to-UAV and UAV to ground station communication to transmit their control, feedback and video signals. From this study, we now know that new control and data channels are required for UAV to UAV and UAV to the ground station communication in the private UAV network to overcome the signals transmission delays. To determine this, we used the 2.4 GHz spectrum with the IEEE 802.11n standard and 5 GHz with the IEEE 802.11ac standard, as they are operating on maximum data rate among other standards and are shown in Fig. 4.13 and Fig. 4.14, respectively. It is important to note that the frequency channels have channel bandwidths measured in MHz, but the data channels have bandwidths or data rates measured in Mbps or Gbps.

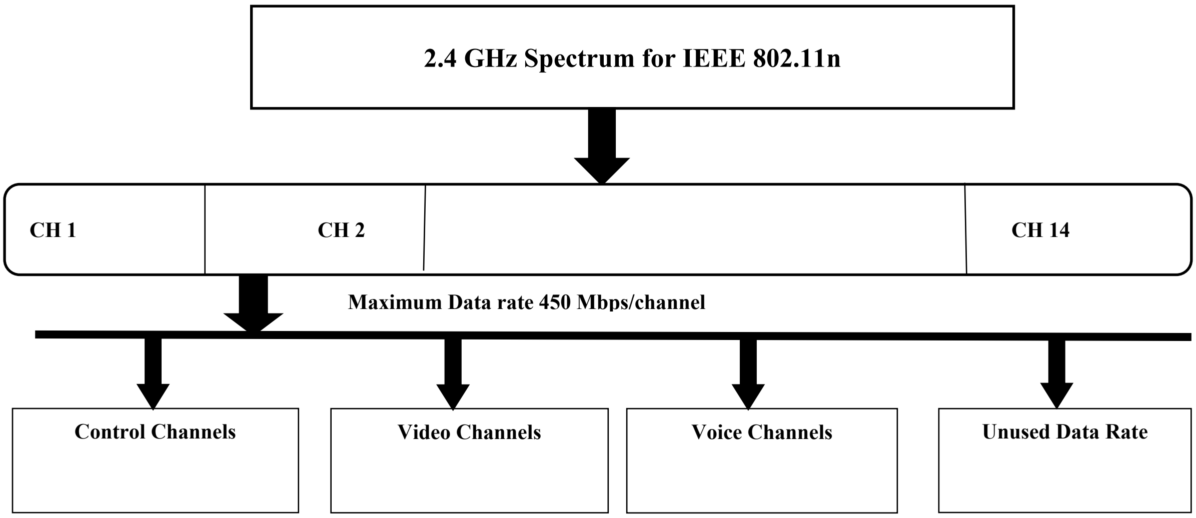


Fig. 4.7 Channel allocation at 2.4 GHz for IEEE 802.11n

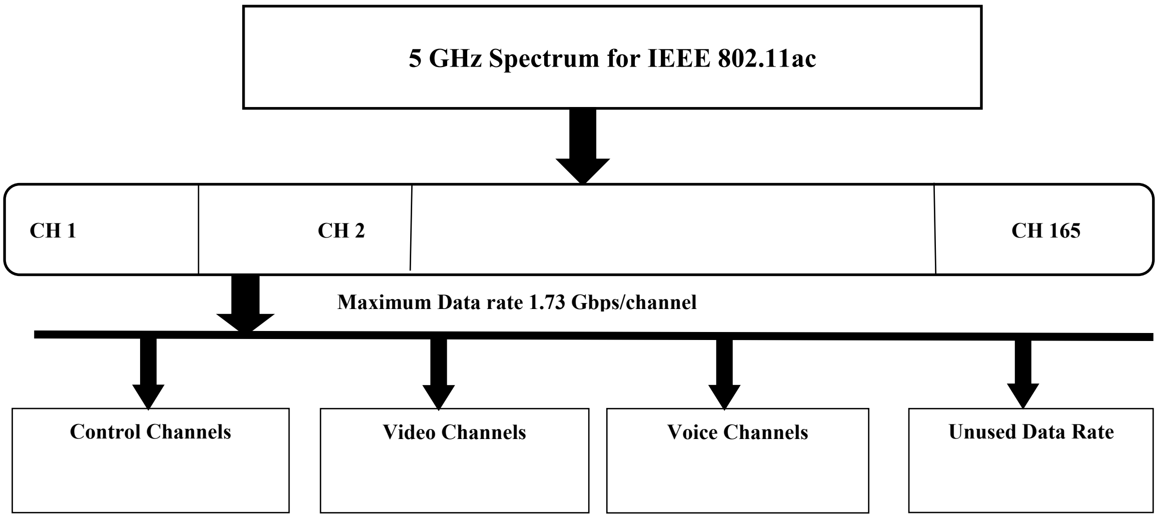


Fig. 4.8 Channel allocation at 5 GHz for IEEE 802.11ac

4.4.1 Bandwidth requirements for transferring videos

As previously mentioned, user and control data have to be transmitted between UAVs and UAV to the ground station in the UAV network. Considering the current applications of commercial UAVs, private UAV networks need to carry many live video streams simultaneously. For any wireless application scenario involving IEEE 802.11 standards, a designer is required to select one allowed channel from the channels provided previously in

Tables 4.5 or 4.6 depending on the frequency spectrum used. If the entire channel is used for communication between two UAVs in the UAV network without separate data channels, control traffic may also be delayed when the traffic load of the user data is high, such as in the case of video transmission in the private UAV network. This issue was investigated during the practical testbed experiments for the private UAV networks. Therefore, after the selection of a frequency channel, the designer of a private UAV network must allocate data channels for various user and control traffic.

To perform data channelisation, we need to understand the minimum data rate required to transmit control signals and videos of varying quality. The standard bandwidth requirement for transmitting HD video is 15 Mbps, but the actual bandwidth requirement depends on the quality of the camera and other parameters. For commercial UAVs currently available on the market, the minimum bandwidth requirement for video transmission varies to some extent. In the DJI Phantom 4 UAV, the live video can be watched with 720p at 30 frames per second on the DJI app. Most small commercial UAVs, such as the DJI Inspire 2, DJI Mavic Pro, DJI Phantom, DJI Spark, Parrot Bebop 2, and Yuneec Typhoon, live video feeds can be sent at 720p with the use of their in-built cameras [8], [77]. Furthermore, most UAV cameras use the H.264 video encoding format, and the required data rate for this video transmission is 1.5 Mbps with 720p at 25 FPS [78]. The data rate requirement for 720p at 30 frames with the H265 codec video transmission also shows that the data of 1800 kbps is sufficient for this purpose (that is, the video resolution requirement) [79]. Given this, we can conclude that with the currently available UAVs that include in-built cameras or cameras recommended by the UAV manufacturers, videos can be transmitted over 2 Mbps data channels. This bandwidth is sufficient for most applications of private UAV networks. Therefore, our channelisation scheme uses a 2 Mbps data rate for video channel creation.

4.4.2 The bandwidth requirement for control channels

A commercial UAV is manoeuvred by a controller which resides in its corresponding ground station. In a private UAV network, control signals travel through several UAV nodes before arriving at the far end UAV node. Therefore, to allocate channels for transmission of the control signals, it is important to know the minimum bandwidth required. For this purpose, we need to investigate the bandwidth allocations used for control channels in other types of known networks. In Integrated Service Digital Network (ISDN), they use 64 kbps for the control signaling [80]. The common channel signaling protocol SS7 also use 64 kbps for control [81]. As such, this data rate should be sufficient to send control signals in the private UAV network.

4.5 Channel creation for the private UAV network

As mentioned in Section 4.3, the IEEE 802.11n and IEEE 802.11ac standards provide the highest data rates for communication between wireless devices, and they operate on the 2.4/5 and 5 GHz spectrums, respectively. Besides this, all commercial UAVs currently available use these two frequency spectrums for communication. Therefore, we created video and control channels to be used in the private UAV network for both of these standards.

4.5.1 Data and control channels with IEEE 802.11n

As previously shown in Table 4.1, the IEEE 802.11n standard can support a data rate of up to 450 Mbps. Since the minimum data rate required for transmitting video traffic is 2 Mbps, as estimated earlier in Section 4.4.1, we can create 218 new video channels with the available data rate of 450 Mbps. These 218 video channels will consume a total of 436 Mbps data rate, and the remaining 14 Mbps data rate can be used to create control channels. Since 64 kbps data rate is allocated for each control channel, 218 control channels can be created. The video and control channel allocation for private UAV network that follows the IEEE 802.11n standard is shown in Table 4.7.

Table 4.7 Channels for the IEEE 802.11n standard

Max. data rate	Data rate requirement for video transmission	Total no. of video channels	Data rate requirement for control signal transmission	Total no. of control channels
450 Mbps	2 Mbps	218	64 kbps	218

The IEEE 802.11n standard can operate on the 2.4 or 5GHz spectrums. Fig. 4.15 shows the channel allocation of IEEE 802.11 n on the 2.4 GHz spectrum.

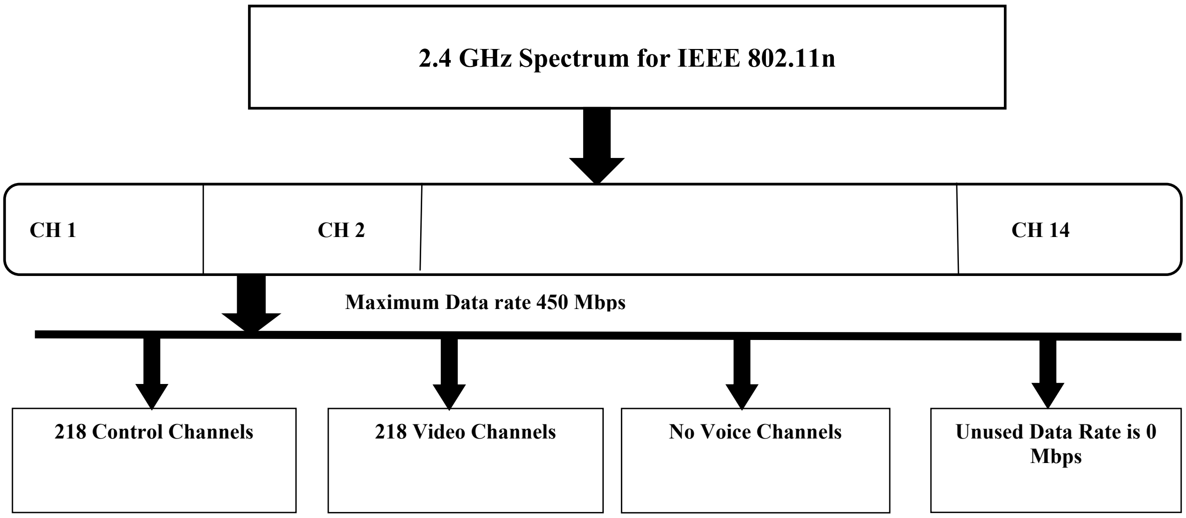


Fig. 4.9 Video and control channels on 2.4 GHz for IEEE 802.11n

This new channelisation is capable of solving the live video streaming delay issue from each UAV to ground station in the private UAV network. We can compare the current channelisation, which we used in our practical testbed with the AR 2.0 UAV network on the 2.4 GHz frequency band, with this new channelisation. As such, with the current channelisation, we have various channels from which we can select a particular channel for UAV-to-UAV communication, but the video, control and feedback data share a single channel. On the other hand, with the new channelisation, we created new channels from a single shared channel and separated the video channels according to their minimum bandwidth requirement. Since video and control signals are transferred via separate channels in this new channelisation scheme in the private UAV network, the problem with delays in video transmission would be successfully addressed with the implementation of these new channels.

4.5.2 Data and control channels with IEEE 802.11ac

IEEE 802.11ac is a powerful wireless communication standard which works on the 5 GHz frequency spectrum. Based on data previously shown in Tables 4.3 and 4.4, this standard can support a data rate of up to 1.73 Gbps when it works on a 160-MHz bandwidth. Since the required data rates for a video and a control channel are 2 Mbps and 64 kbps, respectively, we can create 838 video channels and 838 control channels from the total available data rates. After the creation of these channels, we would still have an unused data rate of 370 kbps. This unused capacity can be used for other types of data such as voice. According to the CISCO [82], a minimum data rate of 20.8 kbps for G.723.1speech coder (5.3 kbps) is sufficient for voice over IP transmission. We can use this data rate and create voice channels from the unused capacity of 370 kbps. The summary of this channelisation is shown in Table 4.8.

Table 4.8 Channels for the IEEE 802.11ac standard

Data rate requirement for video transmission	**Total no. of video channels**	**Data rate requirement for control signal transmission**	**Total no. of control channels**	**Number of voice channels**	**Unused**
2 Mbps	838	64 kbps	838	17	16.4 kbps

Fig. 4.16 shows the video and control channel allocations for IEEE 802.11ac on the 5-GHz spectrum

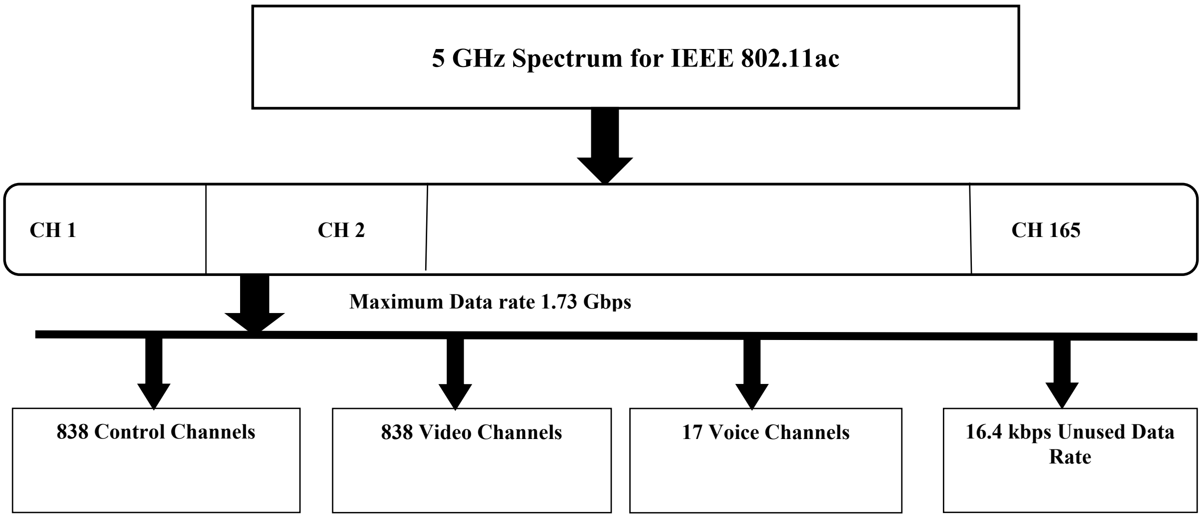


Fig. 4.10 Video and control channels on 5 GHz for IEEE 802.11n

The new channelisation with the 5 GHz private UAV network contains more channels compared with the 2.4 GHz frequency band network. This new channelisation with the 5 GHz network will work in the same way as the 2.4-GHz network, separating the video and control channels to avoid video delay during UAV-to-UAV or UAV to ground station communication in the private UAV network.

4.6 Increasing the channel capacity for video

The new channelisation for the private UAV network allows us to send multiple videos and control signals simultaneously through a UAV network without any interference. As mentioned in Section 4.5, 218 and 834 video and control connections could co-exist in a communication link of a UAV network using the IEEE 802.11n and IEEE 802.11ac standards, respectively.

Table 4.9 Channel bonding with IEEE 802.11n

Number of video channels bonded together	The data rate of one channel (Mbps)	Number of video channels	Number of control channels	Number of voice channels	Unused capacity (kbps)
2	4	110	110	142	6.4
3	6	74	74	60	16
4	8	55	55	311	11.2
5	10	44	44	345	8
6	12	37	37	174	12.8
7	14	31	31	673	17.6
8	16	28	28	10	20.8

In some situations—particularly when UAV camera quality improves into the future—a 2 Mbps channel will not be sufficient to transmit videos of improved quality. For example, to transmit HD quality video, 15 Mbps is needed. For this purpose, eight channels could be bonded together to support HD quality video transmission. Furthermore, the bonding of eight channels would act as a single channel, and we would only need one control channel for all eight channels. In this situation with IEEE 802.11n, the number of video channels would be 28 (that is, each video channel would be a combination of eight channels with a data rate of 2 Mbps), and the number of control channels would be 28 (see Table 4.9). The unused capacity of 208 kbps could then be assigned to ten voice channels.

Similarly, with the IEEE 802.11ac standard, to transmit HD video at 15 Mbps, the number of video and control channels would be 107, with 536 voice channels (see Table 4.10). Tables 4.9 and 4.10 show the number of video and control channels possible with channel bonding to obtain video signals at different capacities.

Table 4.10 Channel bonding with IEEE 802.11ac

Number of video channels bonded together	**The data rate of one bonded channel (Mbps)**	**Number of bonded video channels**	**Number of control channels**	**Number of voice channels**	**Unused capacity (kbps)**
2	4	425	425	134	12.8
3	6	285	285	84	12.8
4	8	214	214	206	19.2
5	10	171	171	435	8
6	12	143	143	233	1.6
7	14	123	123	6	3.2
8	16	107	107	536	3.2

4.7 Comparison of performance of private UAV network with and without channelisation

Using Riverbed modeler academic edition OPNET simulator, we simulated the private UAV network of six UAVs shown in Fig. 4.11. We undertook two simulations, one without channelisation and other one with channelisation.

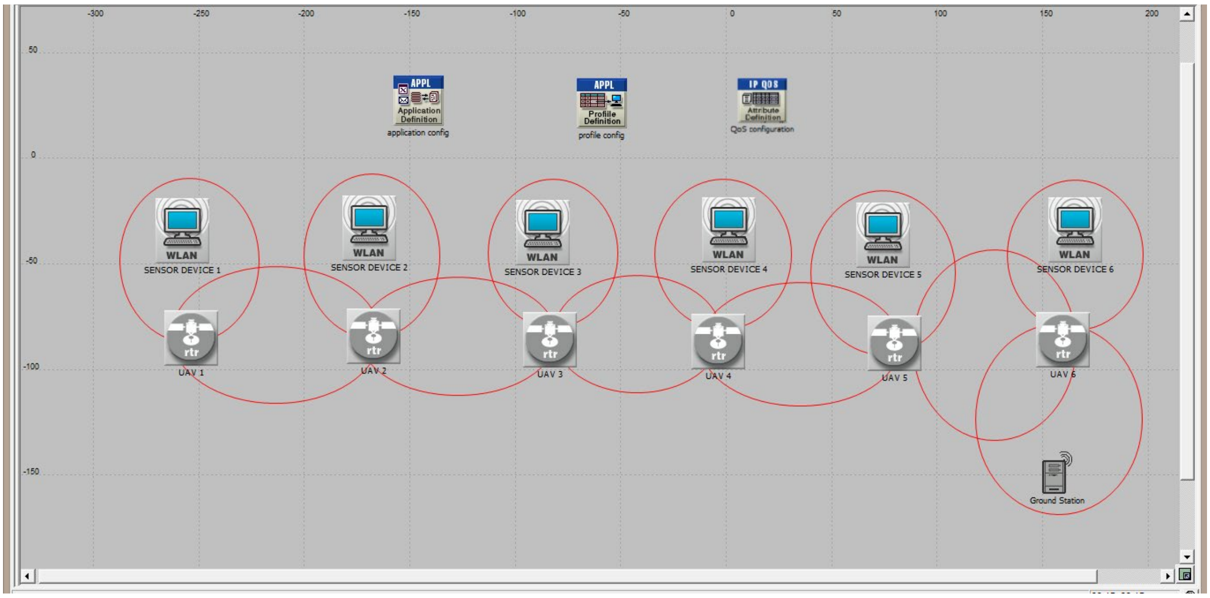


Fig. 4.11 Private UAV network with six UAVs

In the scenario without channelisation, control, feedback and video data are stored in one queue in the WAP and processed. In the second scenario, one queue was allocated for the video traffic and the second queue was used for the control and feedback traffic. The end to end delay at the ground station and packets loss against time were captured and shown in Fig 4.13 and Fig 4.12 respectively.

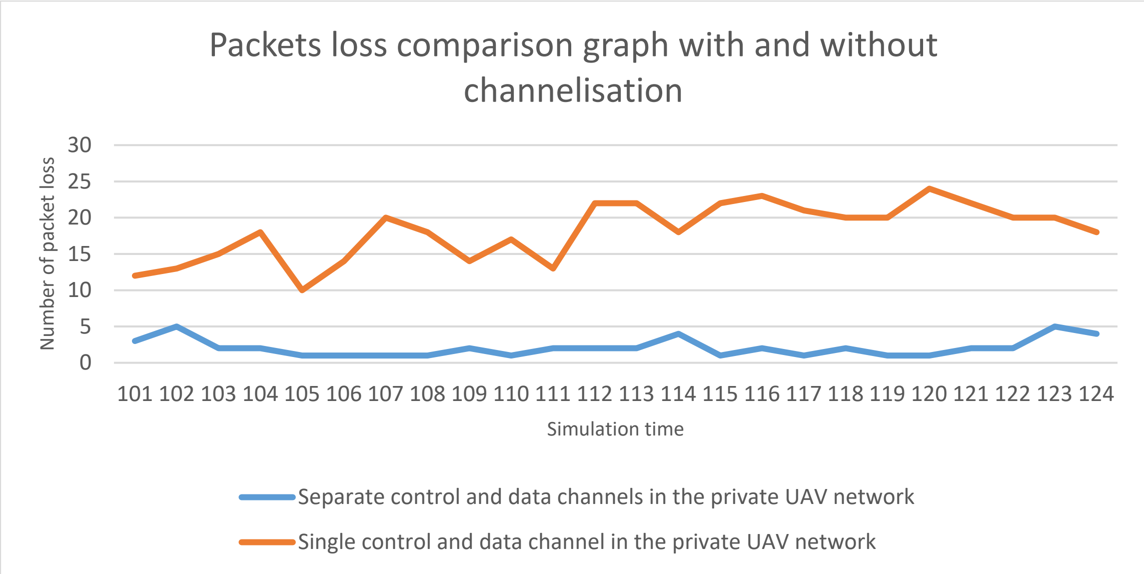


Fig. 4.12 Packets loss comparison for single-channel vs separate channels

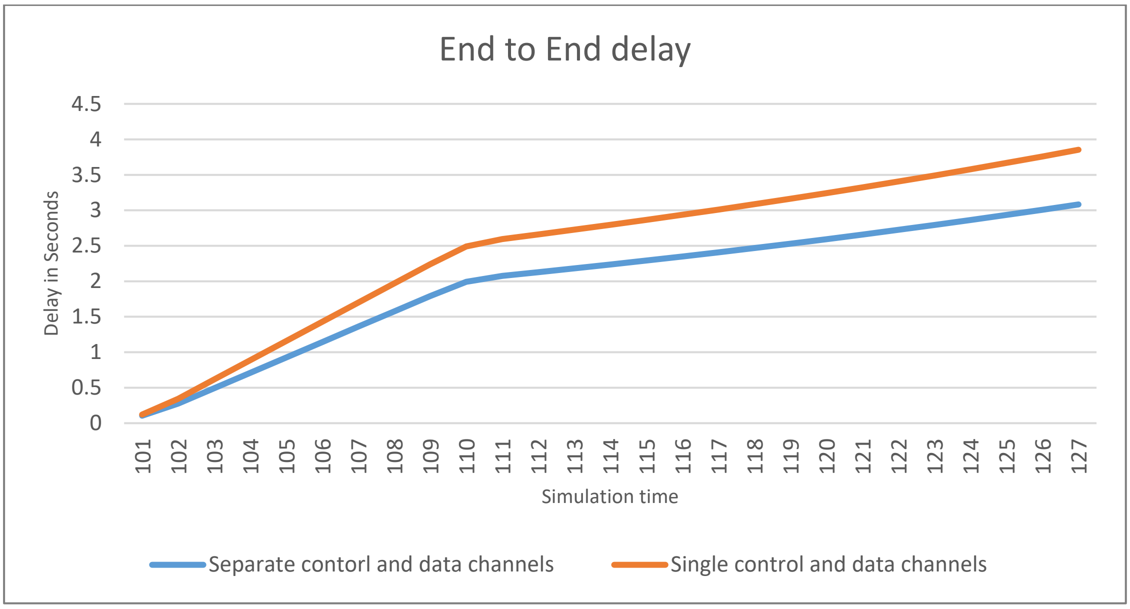


Fig. 4.13 End to end delay comparison for single channel vs separate channel in private UAV network

The traces in Fig. 4.12 and Fig. 4.13 demonstrate that the use of separate channels for video and control traffic have fewer packet loss and less delay compared to the use of a single channel.

4.8 Chapter summary

In this chapter, we presented a new channelisation scheme to transfer control and user data between UAVs and UAV to the ground station in private UAV networks. This new channelisation is required to overcome the delays for delivery of video signals from each UAV to the ground station in a private UAV network. For the development of the new channels, we studied the communication channels and the frequency spectrum used by small commercial UAVs currently available on the market for UAV to ground station communication and reported the minimum bandwidth requirement for video and control data transmission in this network. After investigating the channel allocations on the 2.4 and 5 GHz spectrums, we created new video and control data channels for the IEEE 802.11n and IEEE 802.11ac standards that are suitable to use in private UAV networks to transmit video and control signals from one UAV to another and UAV to the ground station. Furthermore, we presented a solution for high-quality video transmission in private UAV networks in the form of bonding user data channels. We believe the creation of such user- and control data channels are required to avoid packet drop, to improve the control of UAV movements, and, ultimately, to enhance the operation of private UAV networks. We simulated the private UAV network for the new channelisation and compared the result with the single control and data channels for the overall packet loss and the delay at the ground station. The simulation result shows the advantage of new channelisation for the private UAV network.

We proposed and developed the new communication channels that can be used in a private UAV network. With the help of newly developed communication channels, a UAV can transfer its control and data to the other UAV or ground station over the different channels. The performance of a private UAV network also depends on the signaling mechanism used for UAV to UAV and UAV to the ground station communication. Therefore, it is required to analyse the current signaling mechanism in a private UAV network. Hence, in the next chapter, we will discuss the signaling mechanism used in private UAV network.

Chapter 5

Management of signaling protocols for the private UAV network

5.1 Introduction

In Chapter 3, we discussed the implementation of MAC layer communication between UAVs and UAV to the ground station by switching through UAVs nodes to improve the communication in the private UAV network. For this purpose, we implemented two private UAV testbed networks (Storm 4 Mini and AR 2.0 UAV network). In the Storm 4 Mini UAV network, the video signals were transmitted in the Wi-Fi frame format while the analog control signals were used to control this UAV through their remote controller. The feedback signals were not available in this testbed network. In the second AR 2.0 private UAV network, all three signals control, feedback and video were transmitted in the Wi-Fi frame format. Given the implementation of these private UAV networks, it is important to analyse the performance of the current signaling mechanism over the private UAV network. As the AR 2.0 private UAV network is a fully digital signal network. Therefore, we will analyse the signaling mechanism for this private UAV network in this chapter.

In Chapter 3, we already discussed the communication protocols used by AR 2.0 UAV for transmission of its control, feedback and video signals between UAV and the ground station and they were based on the TCP/IP protocol stack. In a private UAV network with AR 2.0 UAVs, when a UAV sends the control, feedback or video signals to other UAV or a ground station, it also uses the same communication protocols as mentioned in Section 3.9. In this network with the current signaling mechanism, a single ground station (computer or smartphone) sends control signals to all of the UAVs, receives video and feedback signals from each UAV. These signals are transmitted between UAVs where two UAVs are directly connected, or through the intermediate UAVs if there is no direct connection between two UAVs in this network. A ground station sends n number of control signals to the first UAV (where n is the number of UAVs in the network); the first UAV processes these signals by extracting the control command and transferring (n-1) control signals to the second UAV. The second UAV does the same thing and transfers control signals to the third UAV and so on. In the same way, feedback and video signals are processed in the private UAV network with the current signaling mechanism. For the feedback and video signals, the last UAV in the network transmits its feedback or video signals to the second last UAVs, second last UAVs send two video and feedback signals to third last UAV, and the ground station receives n number of video and feedback signals. Sending separate control signals to each UAV or receiving separate feedback and video signals from each one of them are inefficient and would lead to waste of channel bandwidth. Depending on the processing power of the UAV and its dynamic memory capacity, this process results in increased packet loss and delay. It then leads to poor control and instability of UAV movements as the UAVs and the ground station will not receive the control and feedback signals on time. Therefore, the control and feedback signal frames used for flying a single UAV has to be studied and should be adapted appropriately to be used in private UAV networks. These signal frames are transmitted through the intermediate UAV nodes in the private UAV network, and each UAV maintains a queue to store these incoming signal frames until they are processed due to its limited processing capability. Therefore, some delay is invariably present when these signal frames are waiting in the queue and processed by the UAV in the network. We used formulas from the queuing theory to analyse these delays at each UAV in this UAV network. Given this, a new way of signaling in the private UAV network is proposed that can overcome these delays. In this new signaling mechanism, the control commands of all UAVs can be sent into a single Wi-Fi frame from the ground station, with the ground station receiving a single feedback frame containing feedback information from all UAVs in the private UAV network. We calculated and compared delays from each UAV to ground station in both signaling mechanisms in this network with the help of queuing theory formulas. These delay calculations in both signaling mechanisms show that the new proposed signaling is the better approach to transfer the signals between UAVs or UAV to the ground station in the private UAV network.

The remainder of the chapter is as follows. We discuss the signaling in a private UAV network in Section 5.2. In Section 5.3, the detailed study of signaling frames is presented. Section 5.4 describes the management of control and feedback frame signaling with the current signaling mechanism and the newly proposed signaling mechanism in the private UAV network. The comparison of the two signaling mechanisms is discussed in Section 5.5, and the chapter is concluded in Section 5.6.

5.2 Signaling in the private UAV network

As discussed in Chapter 3, it was not clear about the network topology for this private UAV network at the early stage of this research, and we connected the UAVs in the tandem. However, another student in the UAV network research group at CQU Sydney campus proved that a private UAV network has a star-connected relay topology, where the ground station is at the head end, and all UAVs are located at the branches [83]. A private UAV network containing three branches is shown in Fig. 5.1. Each UAV in this network has the capability to communicate with its neighbouring UAVs in the same branch. The distance between two UAVs in any two different branches of the private UAV network is larger than the maximum Wi-Fi coverage distance of 100 meters, except where the branches are made to be closer to improve reliability [84].

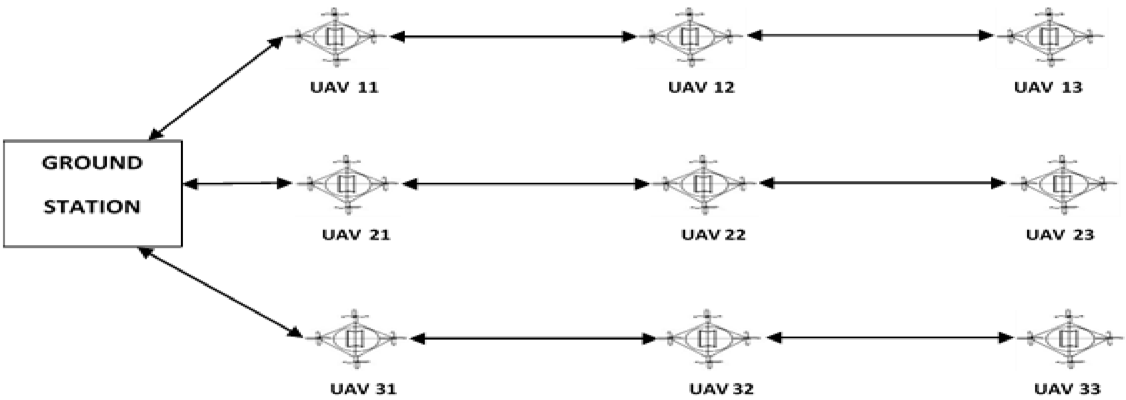


Fig. 5.1 Private UAV network containing three branches

A private UAV network containing only one branch is shown in Fig. 5.2. Each UAV in this network can communicate with its neighbouring UAVs. We will consider this simple private UAV network with one branch, as shown in Fig. 5.2 to study the signaling mechanism.

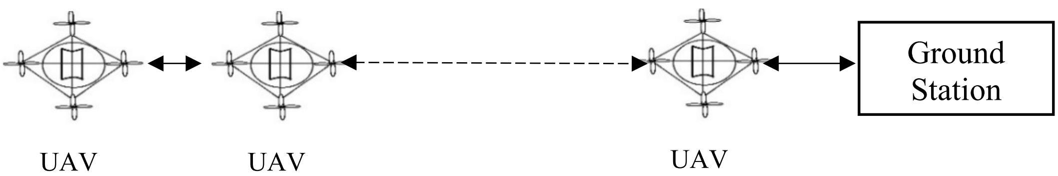


Fig. 5.2 private UAV network with one branch

When a UAV communicates with another UAV in this private UAV network, three types of signals are exchanged: control, feedback and data (that is video or images). These signals follow the same path as shown by the bidirectional arrows in Fig. 5.1 and Fig 5.2.

Furthermore, a single ground station sends the control signals to all the UAVs and receives feedback and video signals from each UAV in this network. In the next section, we start our discussion with the signaling frames that are used to transmit these signals between AR 2.0 UAV and the ground station and then analyse the use of these frames for signaling in a private UAV network.

5.3 Signaling frames used in AR 2.0 UAVs

As discussed in previous chapters, at present, there are two established companies, DJI [9] and Parrot [11] in the market that design a range of UAVs for commercial applications. These two companies work independently and produce different types of UAVs. However, the design of communication signals between the UAV and the controller app has some common features in all these UAVs.

These signals are carried in packets and communication channels of two frequency bands (2.4 or 5 GHz) are used to transfer the control and feedback signals as well as the data signals. In most UAVs, the designers send control and feedback signals in one frequency band, and data is transmitted over the other frequency band to avoid communication interference as well as to minimise the loss of control packets. In such designs, the 2.4 GHz frequency band is used for control and feedback signal transmission, and 5 GHz frequency is used for data transmission. This design is shown in Fig. 5.3. Current designs of Parrot UAVs and some models of DJI are examples of this type of arrangement. However, initial versions of Parrot UAVs such as AR 2.0 uses 2.4 GHz band for both data and control signals.

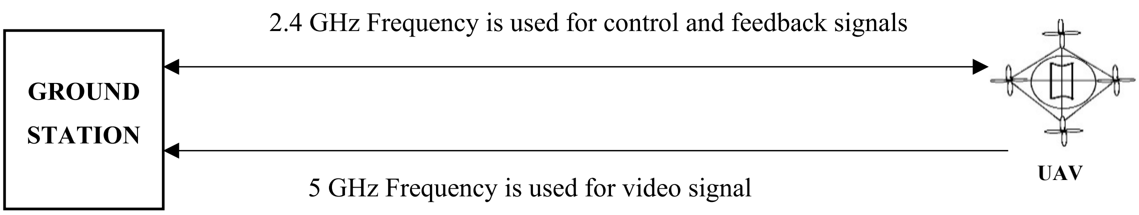


Fig. 5.3 Signaling in the 2.4 GHz frequency.

In some UAVs, these two frequency channels are swapped for transmission of control and data signals. In this type of design, the 5 GHz frequency is used to send control and feedback signals and data is transmitted over the 2.4 GHz frequency—DJI has some UAV models (for example, Phantom 2 Vision and Vision +) which use this convention (Fig. 5.4 illustrates this type of communication arrangement) [8], [10], [11]. We used two different UAV models (that is, the Storm 4 Mini and AR 2.0 models) in the practical testbeds to implement the private UAV network. However, out of these two UAVs, only the AR 2.0 UAV uses the Wi-Fi signal frames to transmit its control, feedback and video signals between UAVs and, therefore, we examined the signaling of this UAV to analyse the signaling in the private UAV network.

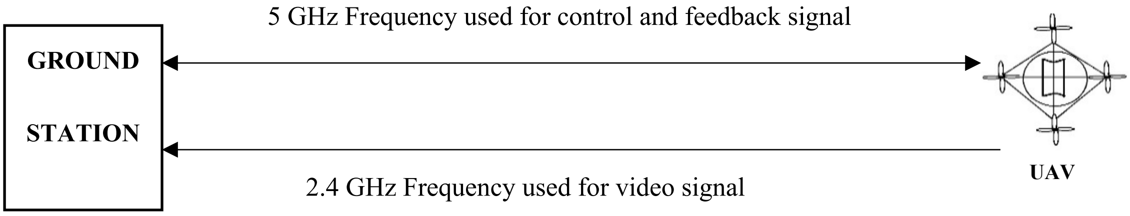


Fig. 5.4 Signaling in the 5-GHz frequency

5.3.1 Parrot AR 2.0 UAV

The Parrot UAV manufacturer company provides a free software development kit and the development guide for the AR 2.0 UAV, which gives details about how control, feedback and data are exchanged between UAV and the ground controller [85]. The AR 2.0 UAV model contains four types of communication services to 1) control the UAV, 2) navigate data, 3) generate live video streams, and 4) control port information. As mentioned in the development guide for the AR 2.0 UAV, once a ground controller is connected with the onboard UAV system, it sends the various control commands to the UAV. This ground controller also receives the feedback, image and video data from the onboard UAV system. The details of these communication conventions are as follows.

5.3.1.1 Controlling and configuration

AR 2.0 UAVs are controlled by sending AT commands on the UDP port 5556. These commands are sent 30 times per second to control AR 2.0 UAVs but at least within two seconds to prevent Wi-Fi connection from being lost from the UAV. A single UDP packet can carry one or more of these commands.

5.3.1.2 Information regarding UAV navigation data (navdata)

This is the second type of communication service of AR 2.0 UAVs. Once the controller connection is established with the AR 2.0 UAV and the control signal is transferred from a smartphone ground controller to the UAV, the UAV sends navdata to the smartphone. The navdata contains information such as the status, speed, engine-rotation speed and battery level of the UAV. In the design of the AR 2.0 UAV model, UDP port number 5554 is allocated to receive this navdata. Furthermore, this information is sent to the client 15 times per second in demo mode and 200 times per second in full mode.

5.3.1.3 The video stream and control port

Depending on the version, AR UAVs send their video data in UDP or TCP packets. UDP port 5555 is assigned to receive the video packets. AR 2.0 UAVs use the TCP protocol, and AR 1.0 UAVs use the UDP protocol for the transmission of video streams, and TCP port 5559 is used to send critical data. Furthermore, this port is used to retrieve configuration information data.

5.3.2 AR 2.0 UAV signaling frames

As discussed previously, the signaling protocols of existing UAVs operate between a UAV and its controller device (that is, the smartphone). In AR 2.0 UAV, the 2.4 GHz frequency is used for the transmission of control and feedback signals. In Chapter 4 of this thesis, the new proposed control, data and voice channels for the private UAV network is presented with the use of both the IEEE 802.11n and IEEE 802.11ac standards. If a private UAV network has this new channelisation then once the ground station is connected with a UAV in the private UAV network, it transmits control and feedback signals over control channels and receives data through data channels. In the following sections, we first examine the different Wi-Fi frames used for transmission between the AR 2.0 UAV and ground station (that is, smartphone or laptop) and then analyse these for the private UAV network.

5.3.2.1 AR 2.0 UAV control frame

Once a controller is bound with an AR 2.0 UAV, it continuously sends control signals to an onboard UAV system. As discussed previously, these control signals are carried in UDP packets in this UAV. In this section, we first look at the UDP packet of the AR 2.0 UAV and then consider how it can be used for signaling in a private UAV network. AR 2.0 UAVs use various control commands to send control signals to an onboard UAV system. These control commands are application data which is generated by a controller (that is, a mobile app or computer). These application data are passed to the transport layer, and a UDP control packet is created. A UDP datagram has two parts: an 8-byte header and a 1024-byte payload as given in Table 5.1. The AR 2.0 development guide provides a list of control commands predominantly used for AR 2.0 UAVs, and these are presented in Appendix B [8].

Table 5.1 AR 2.0 UAV UDP datagram

UDP header	**UDP Data (control commands)**
8 bytes	1024 bytes

Table 5.2 UDP data for control commands

Command	**UDP Data (in bytes)**			
	For command name	**For arguments**	**Carriage return**	**Total UDP data**
AT*REF =	7	8	1	16
AT*PCMD_MAG =	12	28	1	41
AT*FTRIM =	9	4	1	14
AT*CALIB =	9	8	1	18
AT*CONFIG_IDS =	14	16	1	31
AT*COMWDG	9	0	1	10

The details of AR 2.0 UAV control commands are specified in the AR 2.0 development guide. According to this guide, each command starts with three characters AT* then command name. Most of the commands have a number of arguments for a different purpose. The details of these commands are given in Appendix B.2. These commands are transmitted from ground station to the UAV in the UDP packet format and more than one command can be stored in a single UDP packet. However, one control command cannot be split into two UDP packet. The detail of the UDP data for each control command is calculated in Appendix B.2. We studied all AR 2.0 UAV control commands to find the maximum UDP packet size. The AR 2.0 UAV uses some similar types of control commands with a different number of arguments (for example, two commands AT*PCMD and AT*PCMD_MAG). As we were interested in the maximum size of the UDP payload for a single control command, we considered the command with the maximum argument size from among the similar types of commands. Table 5.2 presents the details of the UDP data for each control command.

From Table 5.2, we can see the maximum size of the payload required for the UDP data, as one control command from an AR 2.0 UAV cannot be more than 41 bytes. However, more than one command can be placed in a UDP packet. We used Wireshark to study the control commands of AR 2.0 UAVs. Accordingly, Wireshark traces demonstrated there is a maximum of three control commands in a single UDP packet. Furthermore, even if all of these control commands are sent together, we required only 130 bytes of payload and, therefore, they fit into a single UDP packet. The Wi-Fi frames captured in Wireshark show that a control frame contains 8 bytes of the UDP header, 20 bytes of the IP header and 14 bytes of the Ethernet header. But, the standard IEEE 802.11Wi-Fi frame has 30 bytes MAC header. Furthermore, we investigated and found that capturing these Wi-Fi frames in Wireshark does not work well with Windows and gives incorrect Ethernet header details [86]. Furthermore, we examined Wi-Fi frames in Wireshark with Linux to check the Ethernet header. The Wireshark captured frames in Linux give the correct Ethernet header. Therefore, we used a standard Wi-Fi 802.11 MAC frame format for the AR 2.0 UAV control frame. Table 5.3 shows the maximum size of the control frame for the AR 2.0 UAV (that is, 192 bytes). As verified using Wireshark, a ground station usually sends 30 control commands per second to a UAV.

Table 5.3 Control frame for a private UAV network

802.11 MAC header	**Network data**			**Frame Check Sequence (FCS)**
	IP header	**UDP header**	**Control commands**	
30 bytes	20 bytes	8 bytes	<=130 bytes	4 bytes

5.3.2.2 AR 2.0 UAV feedback frame

Table 5.4 provides information regarding the navdata that is transmitted from the AR 2.0 UAV to the ground station. This navdata is transmitted from the UAV to the ground controller on

UDP port 5554. This navigation data is received periodically (that is, less than 5ms) by the client application and contains information such as flight speed, altitude, distance, roll, pitch, camera and much more. The details of the navigation data are presented in Appendix B. Each UAV sends this information in a single UDP packet. As shown in Table 5.4, standard navdata has a 4-byte header, 4-byte UAV state, 4-byte sequence number, 4 bytes for tag vision, and 4 bytes for the checksum data in the checksum block. The navdata information is stored in several option fields in a UDP packet. Each option field has a 2-byte header, a 16-bit integer for the size of a block, and a data block [85]. These fields are used to specify what type of navigation data is received by the ground station. The information in the option fields is stored in the form of a 32-bit integer and 32-bit single-precision floating-point number or arrays. As in the control frame, the AR 2.0 UAV also contains 4 bytes of a checksum at the end of the frame.

Table 5.4 AR 2.0 UAV navigation data

Header	**UAV state**	**Sequence number**	**Vision flag**	**Option**			**Checksum block**		
32-bit int	32-bit int	32-bit int	32-bit int	Id 16-bit int	Size 16-bit int	Data	Cks id 16-bit int	Size 16-bit int	Cks data 32-bit int

Wireshark showed that each feedback frame has 492 bytes of feedback data, and when we use the standard MAC header, the size of the feedback frame will be 558 bytes. The feedback frame that is transmitted from the AR 2.0 UAV to the ground station is shown in Table 5.5.

Table 5.5 AR 2.0 UAC feedback frame

802.11 MAC header	**Network data**			**Checksum block**	
	IP header	**UDP header**	**Feedback frame data**	**ID and size**	**Checksum**

30 bytes	20 bytes	8 bytes	492 bytes	4 bytes	4 bytes

5.3.2.3 AR 2.0 UAV video frame

AR 2.0 UAV transmits the data signals (that is, video or image) from the UAV to the ground station. The AR development guide does not provide information about the size of this video frame which is generated by the AR 2.0 UAV. We connected the laptop as a ground station with the AR 2.0 UAV and received the live streaming from its onboard cameras. Furthermore, we used Wireshark to analyse the video frame received from the AR 2.0 UAV at the laptop ground station. The AR 2.0 UAV uses a TCP packet for video transmission, and it also confirmed with Wireshark captured frames. However, in the Wireshark captured video frame traces also show the incorrect Ethernet header size, as with the control and feedback frame. Therefore, we used the standard MAC header of IEEE 802.11 to construct the video frame (shown in Table 5.6) with a size of 1534 bytes.

Table 5.6 Video frame of the AR 2.0 UAV

802.11 MAC header	**IP header**	**TCP header**	**Video frame data**	**FCS**
30 bytes	20 bytes	20 bytes	1460 bytes	4 bytes

5.4 Management of control and feedback frames in the private UAV network

In a private UAV network, each UAV can be treated individually, where control messages are simply sent from the ground station to each of them, and the feedback and data messages from each UAV are received at the ground station. However, this will result in wasting bandwidth. Instead, we can send a single control frame to all UAVs in a branch and receive a single feedback frame containing feedback information from all UAVs in a branch. Since a UDP packet has a maximum size, the number of UAVs whose control or feedback information that can be packed into one UDP packet is limited. In Section 5.4.1, therefore, we first investigate this situation and outline the maximum number of UAVs in a branch capable of using a single UDP packet to send control or feedback information.

5.4.1 Single control and feedback frame for each branch

In this section, we discuss our newly proposed signaling mechanism with a single control and feedback frame to manage signaling in a private UAV network. We know that the maximum transfer unit (MTU) in a Wi-Fi network is 2304 bytes [70]. Thus, the maximum size of a UDP packet is 2284 bytes after removing the 20 bytes of IP header. Since the UDP header is 8 bytes, the maximum UDP payload size is 2276 bytes. As we determined before, the maximum payload size required to carry the control signal of an AR 2.0 UAV is 130 bytes, and therefore, we can assemble the control signals of 15 UAVs into a single control packet. On the other hand, the payload size required to carry the feedback signal of a UAV is 500 bytes, including the checksum. Therefore, a single feedback packet can carry the feedback signals of 4 UAVs with 276 unused bytes. If we do not consider the checksum in the feedback data, then also we can have a maximum of 4 UAVs. This arrangement is shown in Fig. 5.5.

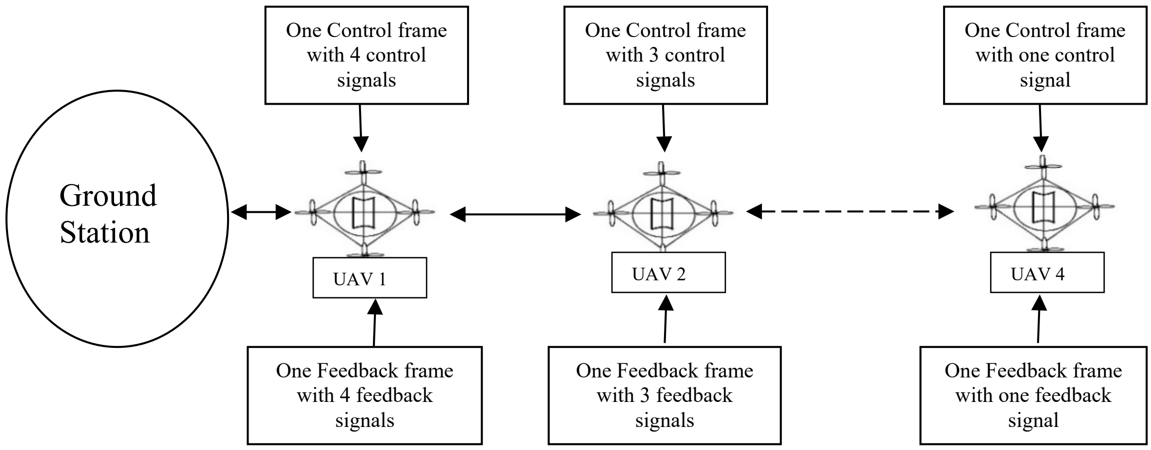


Fig. 5.5 Single control and feedback frames for UAVs in a private UAV network

In a private UAV network with the new signaling mechanism, the ground station can create a single control frame which has the control information for 4 UAVs. This single control frame has a payload of 130*4 = 520 bytes and an 8-byte UDP header, 20-byte IP header, 30-byte 802.11 MAC header and 4-byte FCS (as shown in Table 5.7). In this new single control and feedback signaling for the private UAV network, UAV 1 received a single control frame of 582 bytes; and UAV 1 then processed this frame, extracted its control command and repacked it into a new frame before sending it to UAV 2. UAV 2 then decapsulated the received control frame to extract its control command. This process was repeated at each UAV and continued until the last UAV (that is, UAV 4) received its control commands (as previously shown in Fig. 5.5). Similarly, each UAV processed the feedback frame travelling in the opposite direction from UAV 4 to the ground station. The ground station received a feedback frame with the feedback data of all four UAVs, as shown in Table 5.8.

Table 5.7 The single control frame format for a private UAV network

802.11 MAC header	**IP header**	**UDP header**	**Control data (n represents the UAV number)**	**FCS**
30 bytes	20 bytes	8 bytes	130*n bytes	4 bytes

Table 5.8 Single feedback frame format in a private UAV network

802.11 MAC header	**IP header**	**UDP header**	**Feedback data including ID and size (n represents the UAV number)**	**FCS**
30 bytes	20 bytes	8 bytes	496*(5-n) bytes	4 bytes

In this Table 5.8, n represents the UAV number and we can calculate the feedback frame size of any UAV by replacing n with the UAV number. In this private UAV network, UAV 4 created a feedback frame of 558 bytes and sent it to UAV 3. UAV 3 then processed this incoming feedback frame and constructed a new single frame by adding its feedback data and passing it to UAV 2. The process continued until up to the UAV 1 node. Finally, the ground station receives a single feedback frame that has the feedback information of all UAVs. Table 5.9 shows the control and feedback frame processing at each UAV along a branch.

Table 5.9 Control and Feedback frames processing at each UAV node

Node	**Received control frame size**	**Leaving control frame size**	**Received feedback frame size**	**Leaving feedback frame size**
Ground station		582 bytes	2046 bytes	
UAV 1	582 bytes	452 bytes	1550 bytes	2046 bytes
UAV 2	452 bytes	1322 bytes	1054 bytes	1550 bytes
UAV 3	322 bytes	192 bytes	558 bytes	1054 bytes
UAV 4	192 bytes			558 bytes

From Table 5.9, we can see that the ground station added the control commands of each UAV in a single control frame and each UAV extracted its control command of 130 bytes and reproduced a new control frame for the remainder of the UAVs in the network. In the opposite direction, each UAV added 496 bytes of feedback data to the incoming feedback frame and reproduced a new feedback frame for the next UAVs. Therefore, the ground station receives a single feedback frame that has the feedback information of all UAVs in this private UAV network.

We used only four UAVs in a branch in this private UAV network with this new signaling mechanism. However, in a private UAV network with more than four UAVs in a branch, these single control and feedback frame signaling mechanism can still be used. If the network has more than four UAVs in a branch, a ground station will send one control frame for every four UAVs and receive one feedback frame from those four UAVs. For example, if a branch has eight UAVs, the ground station will send two control frames to first UAV, one for first four UAVs and the other for the last four UAVs. The ground station will receive two feedback frames, one from the first four UAVs and the other from the second 4 UAVs. Each one of this feedback frame has the feedback information of four UAVs.

5.4.2 Separate control and feedback frame per node

The second method of signaling is the transmission of separate control and feedback frames for each UAV, as shown in Fig. 5.6. To draw a comparison with the case of a single control frame, we considered only four UAV nodes in a branch. Accordingly, the ground station generated four control frames (that is, one control frame for each UAV) and then broadcast them. Assuming that only UAV 1 was within its Wi-Fi range, which is the case in a private UAV network, these four control frames will be received by UAV 1 and placed in a queue. The first frame in the queue is processed by UAV 1, and the remaining three control frames are then broadcast again for UAV 2. Next, they will be picked up by UAV 2 and placed in the queue. The process continues until the fourth frame is received at UAV 4. Because the frames for other UAVs have to wait in the queue of a UAV until the frame destined for itself is processed, they incur a delay. In this signaling mechanism, each UAV also generated a feedback frame and broadcast it. As shown in Fig. 5.6, UAV 4 sent its feedback frame to UAV 3, UAV 3 sends feedback frames from itself and UAV 4 to UAV 2, and so on. Finally, the ground station receives four separate feedback frames. These feedback frames also undergo queuing and processing delays. It should be noted that a single queue is used to store the incoming control and feedback frames, but a separate queue is used for video frames.

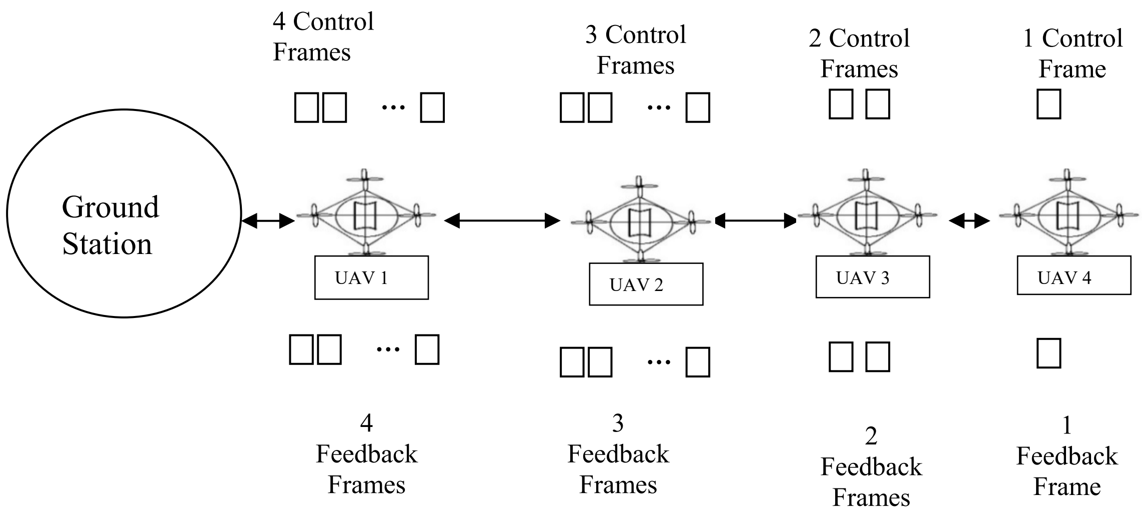


Fig. 5.6 Separate control and feedback frames in a private UAV network

5.5 Analysis of the proposed signaling mechanisms for private UAV networks

In Section 5.4, we discussed two mechanisms for managing the signaling in a private UAV network. In this section, we compare the single control and feedback packet signaling mechanism with the separate control and feedback signaling mechanism. First, calculations of the total delay from the ground station to each UAV for both scenarios used in a private UAV network for this study are given, and then these delays are compared. The total delay to travel a frame from one node to the other in the network depends on several factors. The basic equation for the total delay is given below:

Total Delay (TD) = Processing Delay (PD) + Queuing Delay (QD) + Transmission Delay (TRD) + Propagation Delay (PRD),

where TD is the total delay, and it is the time between the arrival of a frame at a node and arrival of the same frame at the next node. PD is the processing delay and it is the time that a node spends to process the packet. QD is the queuing delay and it is the time that a packet spends in a queue at a node while waiting for the packets ahead of it to be transmitted. In both scenarios, two separate queues are maintained, the first for control and feedback frames and the second for the video frame. PRD is the propagation delay and it is the time that a packet takes to propagate through the communication media from a node to the next node. It is calculated by dividing the distance from the node to the next node by the propagation speed through the medium. As the distance between the two UAVs will always be less than 100 meters, the propagation delay will be the same for both scenarios and we will not consider it in our calculation. TRD is the transmission delay and it is the time required to put an entire frame into the media. TRD depends on the transmission rate. In both scenarios, the transmission rate is the same and as such, we will not consider it in our calculation as well. As such, for comparison purpose, the total delay for both signaling mechanisms will be taken as the sum of PD and QD.

5.5.1 The total delay in separate control and feedback frames

In this section, calculations of the total delay for separate control and feedback frame signaling for a private UAV network are presented. To calculate the delay, we first have to determine the arrival rate (λ) of the control and feedback frames at each UAV, as well as the service rate (μ) of these frames. The service rate refers to the processing speed of the AR 2.0 UAVs to process different incoming frames. Furthermore, the AR 2.0 UAV processes three types of frames (that is, control, feedback and video). Therefore, we have to consider the time-sharing of the AR 2.0 UAV processor to process the control, feedback and video frames, as the processing of video frames always takes more time than the processing of control and feedback frames. The AR 2.0 UAV development guide does not provide information about the processor time-sharing to process these three frames. Since we have already calculated the size of one control and feedback frames and we also know how many of these control and feedback frames can be sent in one second to and from the AR 2.0 UAV, we can then determine the average processing speed to process these frames used by the AR 2.0 processor. In AR 2.0 UAV, for smooth communication with the ground station, a control frame should be sent on a regular basis, usually, 30 times per second. It means AR 2.0 UAV can process 30 control frames per second and, as we have already calculated, the single control frame size of 192 bytes. Therefore, the processing speed required by AR 2.0 UAV to process the control frame will be 46.08 (0.192*8*30) kbps. Furthermore, the AR 2.0 UAV can send 15 feedback frames per second to the ground station, and the size of one feedback frame is 558 bytes. Therefore, the processing speed of the feedback frame is 66.96 (0.558*8*15) kbps. Hence, we need to find out the AR 2.0 UAV video processing rate to calculate the processor time-sharing between control, feedback and video signals. As per the AR 2.0 development guide, the video frame bit rate of this UAV can be set between 500 kbps to 4000 kbps. This is the raw video data rate. We captured many video frames in Wireshark and calculated the average video processing speed. Our calculation gives us the average video frame processing rate of AR 2.0 UAV as 4758 kbps. Therefore, the overall processing speed of AR 2.0 UAV is equivalent to 4871 kbps (46.08+66.96+4758). This processing speed is required to process the control, feedback and video signals. Now, we can calculate the time-sharing of the AR 2.0 processor to process these frames. Hence, in the AR 2.0 UAV, 0.95% (46.08/4871) of the total processing speed is used to process the control frames, 1.37% (66.96/4871) of the total processor speed to process feedback frames and the remaining 97.68% of the total processor speed is used to process the video frames.

In the previous chapters, the requirement of additional WAPs with each UAV to achieve UAV-to-UAV communication in the private network is discussed. Accordingly, to achieve UAV-to-UAV communication, the Edimax WAP is used with the AR 2.0 UAVs in the testbed experiments. The processing speed of this WAP is 150 Mbps. If this WAP capability is provided in each UAV, the overall processing speed of each UAV will be 150 Mbps. Now, we can use the same fractions that we calculated for AR 2.0 to calculate the average processing speeds of control, feedback and video frames in the private UAV network. Therefore, in the private UAV network, each UAV node can process control frames with a speed of 1.43 Mbps (0.95% of 150 Mbps), feedback frames with a speed of 2.06 Mbps (1.37% of 150 Mbps) and video frames with a speed of 146.51 Mbps. As mentioned, the delay calculations are performed using formulas from queuing theory. The control and feedback delay calculation for the separate control and feedback signaling is given in Tables 5.10 and 5.11, respectively. We used the following parameters to calculate these delays.

- λ is the mean rate of arrival of a frame
- μ is the mean service rate or processing speed to process the frame,
- ρ is the utilization of the server
- Lq is the mean number of frames in the queue
- QD is the queuing delay
- PD and TD is the processing delay and total delay respectively
- TDGC is the total delay from the ground station to the UAV node for control frames processing
- TDGF is the total delay from the ground station to the UAV node for feedback frames processing

5.5.1.1 Control frame delay with 4-UAV in a private UAV network

As mentioned earlier, we are considering TD as the sum of QD and PD for this delay calculation and is given by

$$TD = QD+PD \quad (1)$$

QD is calculated with the following formula

$$QD = Lq/\lambda \quad (2)$$

$$Lq = (\rho^2/(1-\rho)) \quad (3)$$

$$\rho = (\lambda/\mu) \quad (4)$$

After substituting ρ in (3) from (4) and Lq in (2) from (3)

$QD = (((\lambda/\mu)^2/ (1- (\lambda/\mu))) /\lambda)$ (5)

$PD = (\lambda/\mu)$ (6)

After substituting QD from (5) and PD from (6), in (1), TD is calculated using the following equation.

$TD = (((\lambda/\mu)2/ (1- (\lambda/\mu))) /\lambda) + (\lambda/\mu)$ (7)

Table 5.10 Total delay for control frame processing

UAV node	**Arrival no. of a control frame**	**The arrival rate of control frames (192 B *30) (λ) (kbps),**	**TD (seconds), where μ = 1.43 Mbps**
1	4	184.320	0.128895208
2	3	138.240	0.096671404
3	2	92.160	0.064447601
4	1	46.080	0.0322238

From Table 5.10, we can see that each UAV receives a different number of control frames. As the size of one control frame is already calculated, and the number of frames which can be processed in one second is also known, we can then calculate the arrival rate (λ) of the control frame at each UAV. Furthermore, as we already know the processing rate of the control frame, we can calculate the TD with the equation (7) for the separate control and feedback signaling mechanism in the private UAV network.

5.5.1.2 Feedback frame delay in the 4 UAVs network

Table 5.11 shows the TD calculation for the processing of the feedback frame in separate control and feedback signaling for the private UAV network. Similar to the processing of the control frame, we first determined the arrival rate of the feedback frame at each UAV by considering the number of arrival feedback frames per second. Once we determined the arrival rate of the feedback frame, we used the equation (7) to discover the TD for the feedback frame processing.

Table 5.11 Total delay for feedback frame processing

UAV node	Arrival no of the feedback frame	The arrival rate of feedback frames (558 B* 15) (λ) (kbps)	TD (seconds), where μ = 2.06 Mbps
1	3	200.880	0.097514616
2	2	133.920	0.065009742
3	1	66.960	0.032504871
4	0	0	0

Table 5.12 The delay from the ground station to each UAV node

UAV node	TD (seconds) for control frames	TDGC (seconds)	TD (seconds) for feedback frames	TDGF (seconds)
1	0.128895208	0.128895208	0.097514616	0.097514616
2	0.096671404	0.225566612	0.065009742	0.162524358
3	0.064447601	0.290014212	0.032504871	0.195029229
4	0.0322238	0.322238012	0	0.195029229

Here we have the delay at each UAV node and, therefore, the delay from the ground station to each UAV can be calculated by adding the delay of all of the UAVs to that particular UAV. For example, if we want to determine the delay from the ground station to UAV 3, we can add the delay of UAV 1, 2 and 3. This, in turn, gives us the delay from the ground station to UAV 3. The overall delay from the ground station to each UAV for control and feedback frames in the private network using the current signaling is given in Table 5.12.

5.5.2 Cross-layer design for single control and feedback frames

In Section 5.4.1, we proposed the single control and feedback frame signaling mechanism for the private UAV network. For this signaling mechanism, we used a cross-layer design to process each incoming frame. In a cross-layer design, the frame is processed at the MAC layer without going through the upper layers of the TCP/IP model. As we used the same type of UAV node with this new signaling mechanism, the same processing speed (μ) is used by the UAV processor to process the control, feedback and video frames. The arrival rate of these frames is different for this new signaling mechanism compared with the current separate control and feedback signaling. In Section 5.3.1, we already calculated the size of the arrival control and feedback frames at each UAV for the new signaling mechanism. As before, the rate of control frames arriving at each UAV and the rate of feedback frames leaving each UAV is 30 frames per second and 15 frames per second, respectively. We used the same queuing delay formulas and calculated the delay for the new mechanism. The calculations of the delay for the control and feedback frames are given in Tables 5.13 and 5.14.

5.5.2.1 Control frame processing of a private UAV network branch with 4 UAVs

From Table 5.13, we can see that only one control frame arrived at each UAV which has the control command for the remainder of the UAVs in the private network.

Table 5.13 Total delay for the single control frame

UAV Node	The arrival rate of control frame per second (λ) (kbps)	TD (seconds), where μ =1.43 Mbps
1	139.680	0.097678397
2	108.480	0.075860197
3	77.280	0.054041998

4	46.080	0.0322238

We used the arrival rate and processing rate of the UAVs to calculate the TD by substituting these values in equation (7) for this new signaling mechanism in the private UAV network.

5.5.2.2 Feedback frame processing of 4 UAVs

Table 5.14 Total delay for the single feedback frame

UAV Node	**The arrival rate of feedback frame (λ) (kbps)**	**TD (seconds) where μ =2.06 Mbps**
1	186.0	0.09029131
2	126.480	0.06139809
3	66.960	0.032504871
4	0	0

Table 5.15 The delay from the ground station to each node

UAV node	**TD (seconds) for control frames**	**TDGC (seconds)**	**TD (seconds) for feedback frames**	**TDGF (seconds)**
1	0.097678397	0.097678397	0.09029131	0.09029131
2	0.075860197	0.173538595	0.06139809	0.1516894
3	0.054041998	0.227580593	0.032504871	0.184194271
4	0.0322238	0.259804392	0	0.184194271

As shown in Table 5.14, we calculated the TD for the feedback frames for the new signaling mechanism. Once we had the TD at each node, we then calculated the total delay from the ground station to each UAV node. Table 5.15 presents the total delay from the ground station to each node for a single control and feedback frame mechanism in the private UAV network.

5.5.3 Delay comparison

In the previous section, the calculations for the delay from each UAV node to the ground station for the control and feedback frames of both signaling mechanisms are presented. Table 5.12 gives the delay from each UAV to the ground station for the current signaling mechanism, where the separate feedback and control frames are transmitted at each UAV. Additionally, as shown in Table 5.15, we calculated the delay from each UAV to the ground station for the single control and feedback signaling mechanism. In Fig. 5.7 (with the help of Tables 5.12 and 5.15), a comparison between the delays is presented in graphical form. From the graph, we can see that our newly developed single control and feedback frame mechanism has less delay compared with the existing signaling process. Percentage calculations of this reduced delay associated with our new signaling mechanism are presented in Table 5.16.

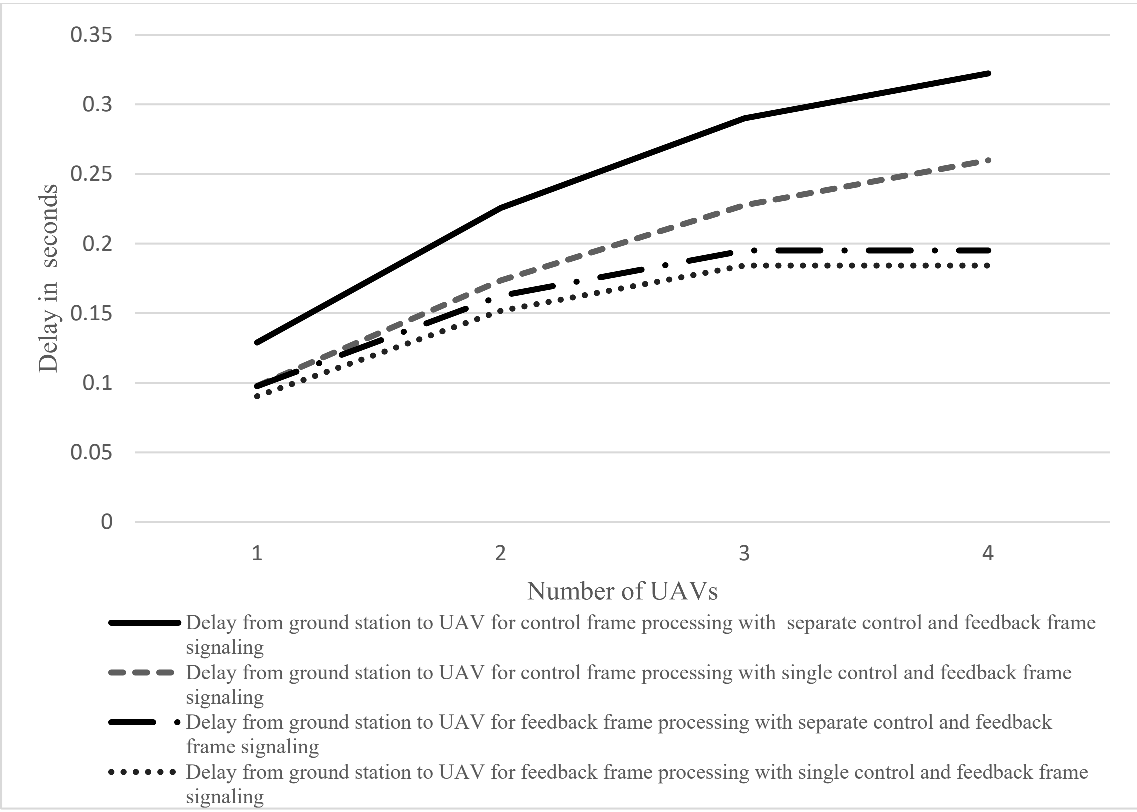


Fig. 5.7 Delay comparison graph

Table 5.16 Reduced Delay in percentage with the new signaling mechanism

UAV node	**TDGC (seconds) in the current signaling mechanism**	**TDGC (seconds) in the newly proposed signaling mechanism**	**Reduced delay (%) for control frames**	**TDGF (seconds) in the current signaling mechanism**	**TDGF (seconds) in the newly proposed signaling mechanism**	**Reduced delay (%) for feedback frames**
1	0.128895208	0.097678397	24.22	0.097514616	0.09029131	7.41
2	0.225566612	0.173538595	23.07	0.162524358	0.1516894	6.67
3	0.290014212	0.227580593	21.53	0.195029229	0.184194271	5.56
4	0.322238012	0.259804392	19.38	0.195029229	0.184194271	5.56

From Table 5.16, we can see that each UAV introduced some delay for the control and feedback frames in both signaling mechanisms for the private UAV network. In this table, calculations of how much the overall delay was reduced (in percentages) by the newly developed single control and feedback signaling for the private UAV network are presented. We can see that our new signaling mechanism reduces the delay around twenty percent for control frame processing and at least five percent for feedback frame processing in the private UAV network. The delay in both signaling mechanisms is due to the high-volume video processing at each UAV in the network. However, our newly developed signaling mechanism helps to overcome this delay. Accordingly, Fig. 5.7 (previously depicting the delays) demonstrates that the single control and feedback frame signaling mechanism is a sophisticated approach in terms of signaling for private UAV networks.

In the private UAV network, frames have to pass from one UAV node to the other. Thus, if a node malfunctions, frame corruption happens whether we have a single control and feedback packet or not. The reliability under node malfunctioning has already been discussed in [84]. The research result in [84] shows that two redundant branches will give near 100% reliability.

The packet loss rate for the AR 2.0 UAV is not given or measured in the literature. However, Silva et al.[87] have conducted multiple experiments to evaluate the performance of Multi-UAV network. They calculated the packet loss by sending 500 packets between UAVs in the Multi-UAV network. This research result shows a maximum of 0.9 packet loss out of 500 packets transmitted between UAVs in different scenarios. We can consider this packet loss rate to estimate the packet loss rate of the network with the newly proposed signaling mechanism. Hence, the frame loss rate with separate control and feedback signaling for each UAV is given by 0.18% (0.9/500*100). Since we are combining the control or feedback signals of four UAVs into a single frame, the frame loss rate will be 0.72% and is very low. As such, the reduction in delay comes at the cost of increasing the frame loss rate.

5.6 Simulation Study for delay reduction with new signaling mechanism

We simulated a private UAV network of four UAVs and studied the end to end dealy reduction and packet loss. This was undertaken by changing the distance between UAVs and adding extra traffic loads for the cases of single control and feedback frame signaling as well as separate contorl and feedback frame signaling.

5.6.1 Single versus separate control and feedback frame signaling with distance change

In this simulation on OPNET, we created a private UAV network with four UAV nodes and configured similar to the network used in section 5.3. The four UAVs were configured to be 20m apart from each other for the first simulation and changed to 40m for the second simulation as shown in Fig. 5.8 and Fig. 5. 9 respectively.

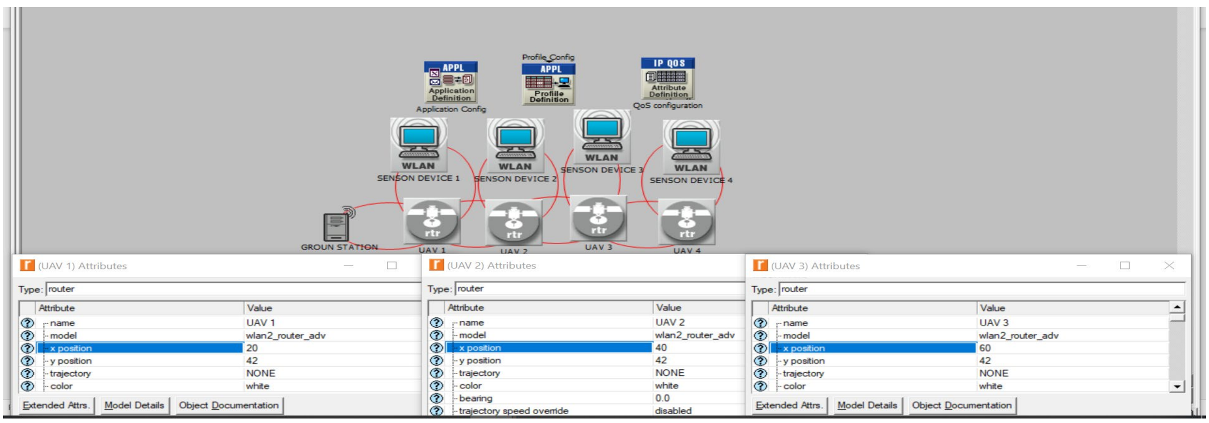


Fig. 5.8 Four UAVs private UAV network with 20m distance between UAVs

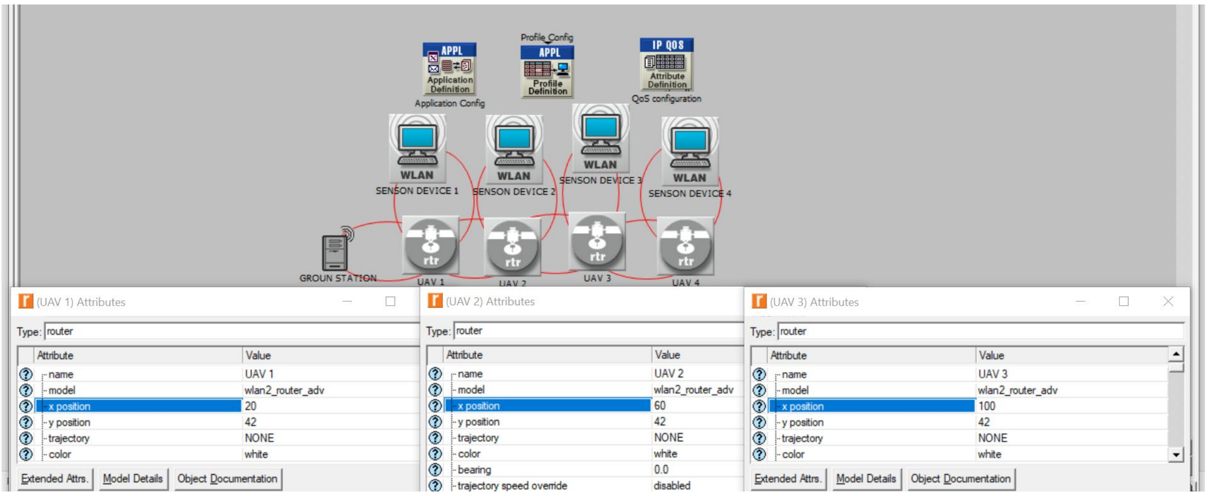


Fig. 5.9 Four UAVs private UAV network with 20m distance between UAVs

In the scenario with separate control and feedback frame signaling, each WAP processed separate control and feedback signals that were generated by each UAV in the network. In the second scenario with the proposed single control and feedback frame signaling, each WAP processed a single control and feedback frame. The end to end delay and the packet loss at the ground station were captured for the duration of simulation and are shown in Fig 5.10 and Fig 5.11 respectively.

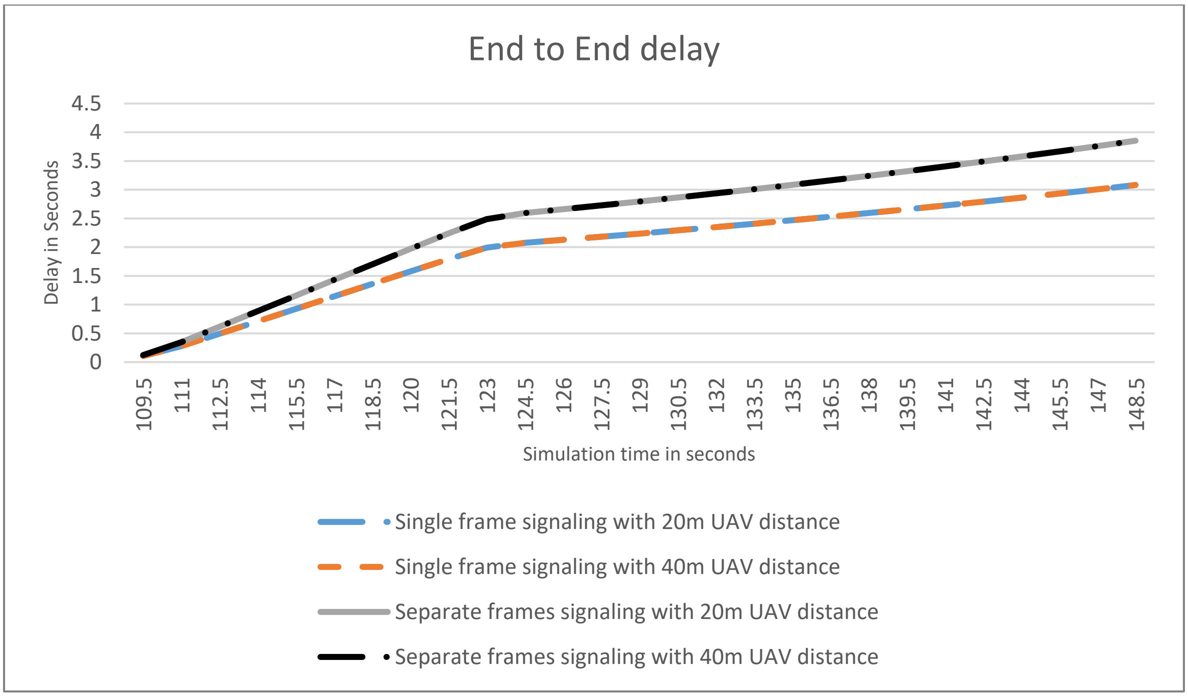


Figure 5.10 End to end delay comparison for single and separate control and feedback frame signaling

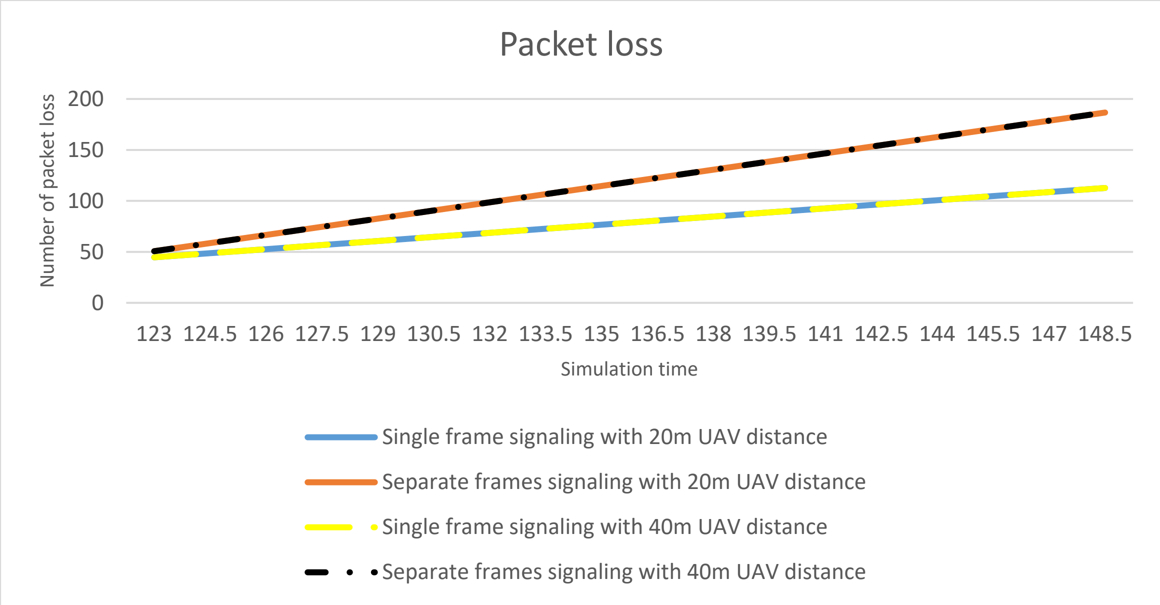


Fig. 5.11 Packet loss at the ground station for single and separate control and feedback frame signaling

Fig. 5.10 and Fig. 5.11 prove that there is no effect of the distance change between UAVs on the communication as long as UAVs are in the 100m communication range of each other. Therefore, we can conclude that the proposed single control and feedback frame singling mechanism is a better efficient approach for signaling in private UAV networks.

5.6.2 Single versus separate control and feedback frame signaling with different traffic loads

To study the effect of traffic load on the performance in the cases of single as well as separate contorl and feedback frame signlaing, we added traffic loads for each UAV one by one and compared the end to end delay and packets loss for the two cases.

The end to end delay at the ground station and packet loss were captured for different traffic loads for both cases and shown in Fig 5.12 and Fig 5.13 respectively.

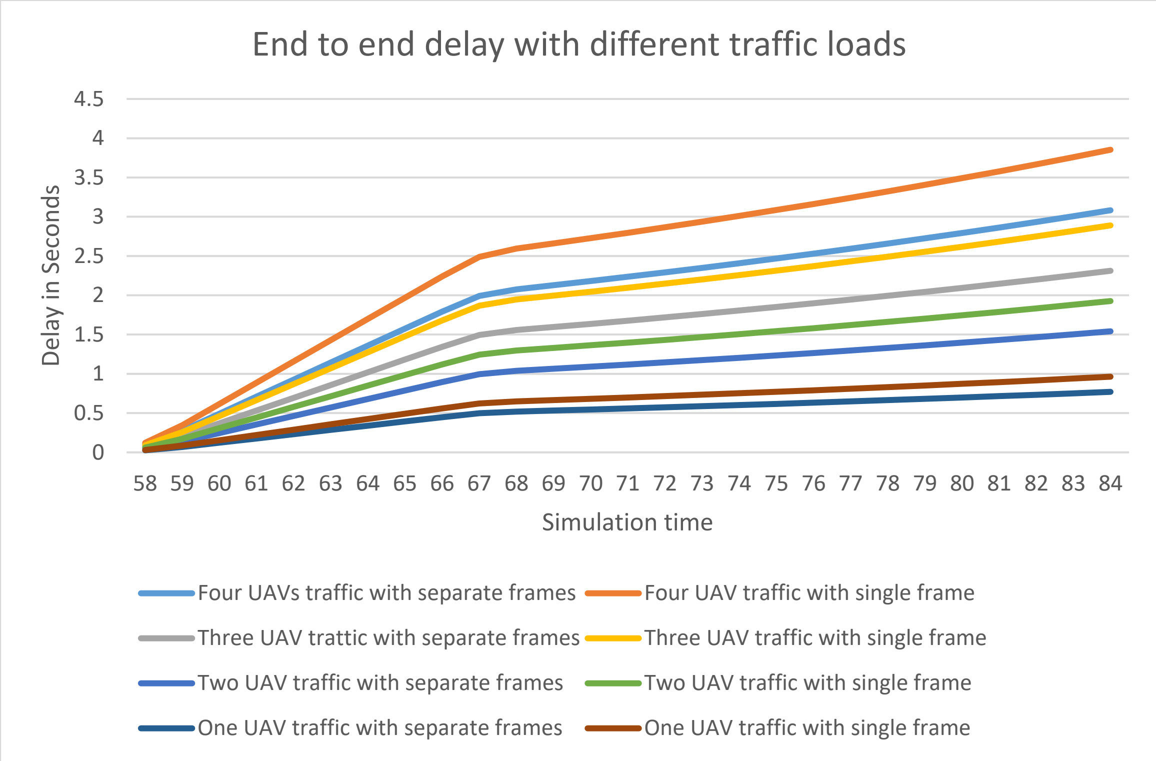


Figure 5.12 End to end delay comparison graph with different traffic load for the signaling mechanism

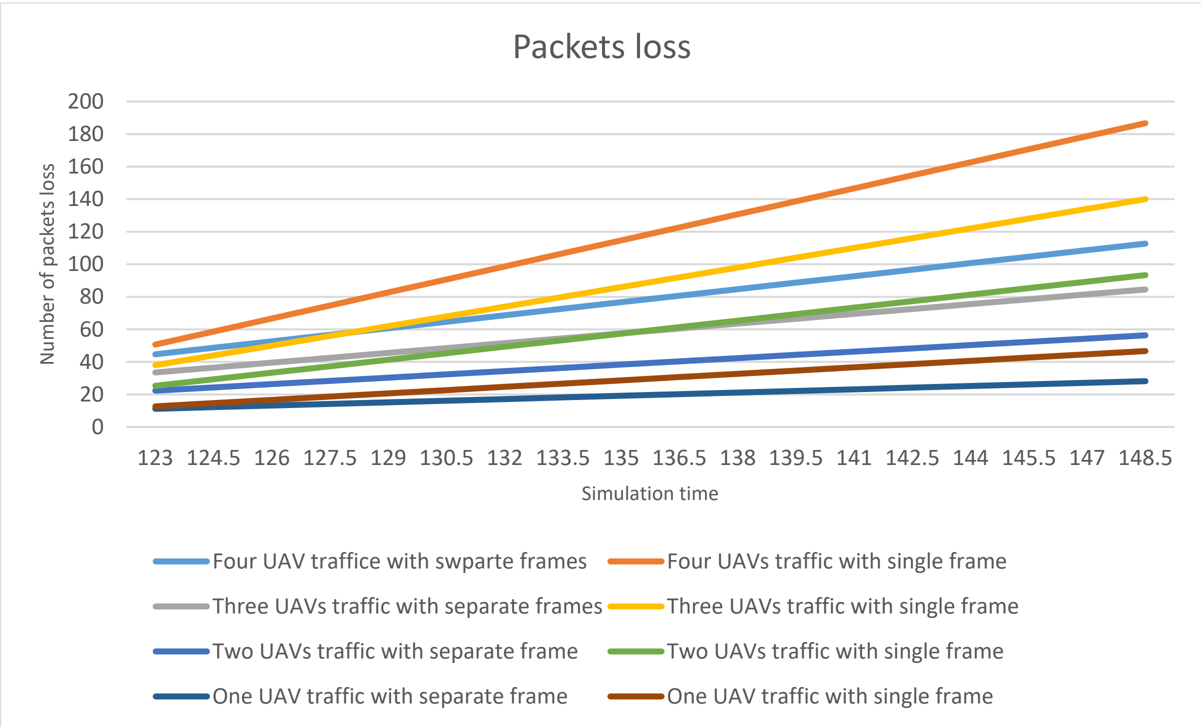


Fig. 5.13 Packets loss comparison with different traffic loads for the signaling mechanism

We can see from Fig. 5.12 and Fig. 5.13 that the end to end delay and packets loss at the ground station have incresed by incresing the traffic load on the network with both signaling mechanisms. But in all cases, the packet loss and end to end delay at the ground station are less for the proposed single control and feedback signaling mechanism. We can conclude from these simulation results that the single control and feedback signaling mechanism is a better approach for signaling in private UAV networks.

5.7 Chapter summary

In this chapter, we studied the current signaling mechanism in AR 2.0 private UAV network and proposed a new signaling mechanism to reduce the control and feedback transmission delay between the ground station and each UAV. For this purpose, we studied the control, feedback and video frame rates of AR 2.0 UAV that is transmitted between a UAV and ground station. In the private UAV network, when multiple frames arrive at a single UAV node, they have to wait in the queue before being transmitted to the next UAV node, and this introduces a delay in communication. However, each UAV in the private UAV network can maintain two separate queues, one for control and feedback frames and the other for video frames to mitigate the queuing delay. Using a cross-layer design and combining the payloads of frames destined for 4 UAVs in a private UAV network branch into single frames, the delay experienced by control and feedback frames could be reduced to around twenty percent and at least five percent respectively. Therefore, this new signaling mechanism is a better approach for the private UAV network. We tested the newly proposed signaling mechanism with simulation and compared the delay and packet loss with the current signaling mechanism by changing the distance between UAVs and increasing the traffic load. The simulation results show that the newly proposed signaling mechanism has less delay and it is a better approach for signaling in the private UAV network. In future, the commercial smartphone operated UAVs can use this new signaling approach while designing the new UAV models that will help to enhance the performance of the private UAV network.

Chapter 6

Summary and discussion

6.1 Summary

In this research, we implemented the communication between UAVs and UAV to the ground station in the private UAV testbed networks at the MAC layer by switching through UAV nodes. We successfully tested these communications in the private UAV network for fast data transmission between UAVs and UAV to the ground station. This study improved the communication in a private UAV network, with particular focus on the management of signaling between UAVs for this network. In Chapter 2, we discussed the UAVs and their application in the various applications domain. Since a UAV network can give better services for these commercial applications, we discussed the existing UAV networks as well as a ground-based ad-hoc network that has some similar characteristics with FANET in this chapter. We also studied the existing communication protocols used in these networks and discussed them in this chapter. A piece of deep information about the private UAV network was presented in Chapter 3 for use in commercial UAV applications. Furthermore, we outlined the simulation of private UAV network (that is, with the use of the OPNET simulator) to test the UAV to UAV and UAV to the ground station communication in this chapter. Also, in Chapter 3, the implementation of two experimental private UAV testbed networks was presented to test the MAC layer communication between UAVs and UAV to the ground station in this study. In these testbed experiments, it was revealed that the use of a common single communication channel for transmission at each UAV caused considerable delays in live video streaming at the ground station. Therefore, we developed new channelisation for the private UAV network in this study to address this problem, which was outlined in Chapter 4. In Chapter 5, a new signaling mechanism was proposed, which resulted from an examination of the current signaling mechanism in the private UAV network. The newly proposed signaling mechanism overcomes the signal transmission delay from each UAV to the ground station in the private UAV network as compared to the current signaling mechanism.

6.2 General discussion

In this study, we first implemented the communication between UAVs and UAV to the ground station in a private UAV network at MAC layer by switching through UAV nodes and all the UAVs in this network could send their data to a single ground station. Following research on the commercial UAVs currently available on the market, we found that they cannot communicate with other UAVs because they were designed to be controlled from their controlling device. Therefore, we attached an additional configurable WAP with each UAV and gave them the functionality to communicate with other UAVs and formed a private UAV network with six UAVs. We first tested the communication in this private UAV network with the help of the OPNET simulator. Secondly, we implemented two practical testbeds (that is Storm 4 Mini UAV and AR 2.0 UAV) to demonstrate the real-time communication inherent in this private UAV network. In doing this, we encountered problems with delay in terms of receiving video data at the ground station used in the study. We then examined these practical testbed networks and discovered that, due to a single common channel used for transmission of control, feedback and data, this delay was introduced at each UAV node. To address this issue, we proposed new channelisation for this private UAV network in which we separated the control and data channels. As UAVs are controlled by control signals transmitted from the ground station, we examined how UAV to ground station signaling occurs in single UAVs (that is AR 2.0 UAV). After obtaining information regarding the communication protocols for a single AR 2.0 UAV, we then analysed the private UAV network with the current signaling mechanism used by AR 2.0 UAVs. Accordingly, we calculated the delay associated with current signaling to transmit the control and feedback signals from each UAV to the ground station in the private UAV network and then proposed a new single control and feedback signaling mechanism to overcome it.

6.3 Limitations

The current study includes several limitations. First, we tested the communication between UAVs and UAV to the ground station in the final practical testbed of Storm 4 Mini network without flying and we only used three in flying AR 2.0 UAVs private UAV network. Specifically, we tested a single Storm 4 Mini UAV and found that it could hold the payload of WAP and a Wi-Fi camera. We also did some testing of the AR 2.0 UAV network with indoor flying; however, we did not test the complete network communication system with six UAVs of both private UAV networks in flight due to limitations regarding space. Since the main purpose of these experimental testbed networks was to test the communication in the private UAV network at the MAC layer and it was successfully performed.

The second limitation is related to the batteries of the UAV model. Consequently, when these UAVs were in flight as part of the network, each UAV battery needed to be either recharged or their battery changed frequently. This is another area of research to increase the flying time of commercial UAV models without losing its payload capacity.

The third limitation with the new channelisation, we proposed and developed new channelisation, but we were unable to test this new channelisation in the private UAV network. We used the Edimax WAP to implement the communication between two UAVs in this private UAV network, which operates on the 2.4- and 5-GHz frequency bands. These frequencies channels are standardised, and we were unable to change these channels.

Furthermore, we proposed a new signaling mechanism for our private UAV network but were unable to test it. Specifically, the UAV models were designed to send their control and feedback signals with its own ground station only. We could not change the design of the existing UAVs to combine the multiple UAVs control and feedback signals.

6.4 Future directions

Several issues pertaining to communication between UAVs and UAV to the ground station in the private UAV network were addressed in this study. In the future, UAV manufacturers should design UAV models capable of communicating with a ground station as well as with another UAV by considering the practical problems and solutions proposed in this research. The new system of channelisation proposed in this study also offers the opportunity for research in the future. Specifically, the redesigning of the UAV control system is required to implement this new signaling mechanism, and this presents avenues for new studies. We worked on the control and feedback signals with the AR 2.0 UAV to develop the new signaling mechanism, but there is the requirement of redesigning a new control system with this new signaling mechanism for the UAVs that can be used to form a private UAV network to improve the performance of this network for the commercial application of UAVs. Data Security in the private UAV network is also an important factor. When a UAV transmits its data to another UAV or the ground station in the private UAV network, these data transmission should be secure. However, it is a different area of research.

6.5 Conclusion

Several conclusions can be drawn from the current study. First, the UAV mode with additional WAP was worked well to test the communication between UAVs and UAV to the ground station at MAC layer in the private UAV network. Second, the MAC layer communication between UAVs and UAV to the ground station improves the overall communication in the private UAV network. Furthermore, the new channelisation proposed achieved a significant reduction in video delay at the ground station in this network. Finally, it was revealed in this study that signaling delays can be overcome by adapting the newly proposed signaling mechanism.

Overall, the commercial application of UAV required a private UAV network to provide better service to the individual or small organisation. The backbone of this private UAV network is its communication system. We implemented a better communication system for the private UAV network and tested it successfully for UAV to UAV and UAV to the ground station communication. In this MAC layer communication for the private UAV network, several practical issues encountered and solutions were provided during the implementation of small testbed networks in this study. We also discussed the improvement of fast signal transmission between UAVs and UAV to the ground station by purposing new channelisation and signaling mechanism for this private UAV network.

References

[1] CB Insights, "38 Ways Drones Will Impact Society: From Fighting War To Forecasting Weather UAVs Change Everything", [Online]. Available: https://www.cbinsights.com/research/drone-impact-society-uav/. [Accessed on :15-Feb-2019].

[2] L. Changchun, S. Li, W. Hai-bo, and L. Tianjie, "The research on unmanned aerial vehicle remote sensing and its applications," in *2010 2nd International Conference on Advanced Computer Control*, 2010, vol. 2: IEEE, pp. 644-647.

[3] Y. M. Chen, L. Dong, and J.-S. Oh, "Real-time video relay for uav traffic surveillance systems through available communication networks," in *2007 IEEE wireless communications and networking conference*, 2007: IEEE, pp. 2608-2612.

[4] B. Stevenson, "DHL to test supply delivery using UAVs", Flight Global [Online]. Available: https://www.flightglobal.com/news/articles/dhl-to-test-supply-delivery-using-uavs-404143/. [Accessed on : 10-Mar-2016].

[5] J. Primicerio *et al.*, "A flexible unmanned aerial vehicle for precision agriculture," *Precision Agriculture,* vol. 13, no. 4, pp. 517-523, 2012.

[6] G. Mezzofiore, "New lifesaving drone rescues swimmers by dropping inflatable pod", CNN, [Online]. Available: https://edition.cnn.com/2018/01/18/tech/drone-rescue-swimmers-australia/index.html. [Accessed on : 5-Mar-2018].

[7] Surf Life Saving NSW, "Port Macquarie lifeguards use drones to assist emergency services with bushfire", [Online]. Available: https://www.surflifesaving.com.au/news/port-macquarie-lifeguards-use-drones-assist-emergency-services-bushfire. [Accessed on : 5-Nov-2018].

[8] DJI, "DJI UAV Model", [Online]. Available: https://www.dji.com/au. [Accessed on : 5-Mar-2017].

[9] Helipal, "Storm drone 4", [Online]. Available: http://www.helipal.com/storm-drone-4-flying-platform-v2.html. [Accessed on :2-Jan-2016].

[10] Parrot, "Parrot UAVs", [Online]. Available: https://www.parrot.com/global/. [Accessed on:4-Feb-2017].

[11] 3DR, "3DR Solo Drone", [Online]. Available: https://3dr.com/solo-drone/. [Accessed on :10-Aug-2016].

[12] I. Bekmezci, O. K. Sahingoz, and S. Temel, "Flying ad-hoc networks (FANETs): A survey," *Ad Hoc Networks,* vol. 11, no. 3, pp. 1254-1270, 2013.

[13] M. Gharibi, R. Boutaba, and S. L. Waslander, "Internet of drones," *IEEE Access,* vol. 4, pp. 1148-1162, 2016.

[14] R. de Silva, "Private UAV Networks and LANs of Drones," *ed: Leisure Informatics Society 2018,* [Online]. Available: https://www.leisureinformatics.com/t31-private-uav-networks-and-lans-of-drones-by-rohan-de-silva-10-01-2018-page-2-4.

[15] M. E. Dempsey and S. Rasmussen, "Eyes of the army–US Army roadmap for unmanned aircraft systems 2010–2035," *US Army UAS Center of Excellence, Ft. Rucker, Alabma,* vol. 9, 2010.

[16] P. Cohn, A.Green, M. Langstaff, and M.Roller, "Commercial drones are here: The future of unmanned aerial systems", McKinesey & Company, [Online]. Available: https://www.mckinsey.com/industries/capital-projects-and-infrastructure/our-insights/commercial-drones-are-here-the-future-of-unmanned-aerial-systems. [Accessed on:15-May-2019].

[17] M. Hassanalian and A. Abdelkefi, "Classifications, applications, and design challenges of drones: A review," *Progress in Aerospace Sciences*, vol. 91, pp. 99-131, 2017.

[18] S. M. Adams and C. J. Friedland, "A survey of unmanned aerial vehicle (UAV) usage for imagery collection in disaster research and management," in *9th International Workshop on Remote Sensing for Disaster Response*, 2011, vol. 8.

[19] M. Abdelkader, M. Shaqura, C. G. Claudel, and W. Gueaieb, "A UAV based system for real time flash flood monitoring in desert environments using Lagrangian microsensors," in *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2013: IEEE, pp. 25-34.

[20] G. Tuna, B. Nefzi, and G. Conte, "Unmanned aerial vehicle-aided communications system for disaster recovery," *Journal of Network and Computer Applications,* vol. 41, pp. 27-36, 2014.

[21] C. Lelong, P. Burger, G. Jubelin, B. Roux, S. Labbé, and F. Baret, "Assessment of unmanned aerial vehicles imagery for quantitative monitoring of wheat crop in small plots," *Sensors,* vol. 8, no. 5, pp. 3557-3585, 2008.

[22] F. G. Costa, J. Ueyama, T. Braun, G. Pessin, F. S. Osório, and P. A. Vargas, "The use of unmanned aerial vehicles and wireless sensor network in agricultural applications,"

in *2012 IEEE International Geoscience and Remote Sensing Symposium*, 2012: IEEE, pp. 5045-5048.

[23] J. Rennie and H. Chelard, "The use of unmanned aerial vehicles (UAVs) in agriculture - regulations, challenges and opportunities." [Online]. Available: https://grdc.com.au/resources-and-publications/grdc-update-papers/tab-content/grdc-update-papers/2015/02/the-use-of-unmanned-aerial-vehicles

[24] M. Quigley, M. A. Goodrich, S. Griffiths, A. Eldredge, and R. W. Beard, "Target acquisition, localization, and surveillance using a fixed-wing mini-UAV and gimbaled camera," in *Proceedings of the 2005 IEEE international conference on robotics and automation*, 2005: IEEE, pp. 2600-2605.

[25] E. Semsch, M. Jakob, D. Pavlicek, and M. Pechoucek, "Autonomous UAV surveillance in complex urban environments," in *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology-Volume 02*, 2009: IEEE Computer Society, pp. 82-85.

[26] A. W. N. Ibrahim, P. W. Ching, G. G. Seet, W. M. Lau, and W. Czajewski, "Moving objects detection and tracking framework for UAV-based surveillance," in *2010 Fourth Pacific-Rim Symposium on Image and Video Technology*, 2010: IEEE, pp. 456-461.

[27] F. Rinaudo, F. Chiabrando, A. Lingua, and A. Spanò, "Archaeological site monitoring: UAV photogrammetry can be an answer," *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* vol. 39, no. B5, pp. 583-588, 2012.

[28] R. Burgur, "6 Ways Drones Are Affecting the Construction Industry The balance small business", [Online]. Available: https://www.thebalancesmb.com/drones-affecting-construction-industry-845293. [Accessed on :10-Oct-2019].

[29] S. Millar, "Drone photography changing the face of real estate", Real Estate, [Online]. Available: https://www.realestate.com.au/news/drone-photography-changing-the-face-of-real-estate/. [Accessed on : 17-Mar-2017].

[30] S. K. Sarkar, T. G. Basavaraju, and C. Puttamadappa, *Ad hoc mobile wireless networks: principles, protocols, and applications*. CRC Press, 2016.

[31] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, and H. Zedan, "A comprehensive survey on vehicular ad hoc network," *Journal of network and computer applications,* vol. 37, pp. 380-392, 2014.

[32] S. Eichler, C. Schroth, and J. Eberspächer, "Car-to-car communication," 2006.

[33] Y. J. Li, "An overview of the DSRC/WAVE technology," in *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, 2010: Springer, pp. 544-558.

[34] C.-M. Cheng, P.-H. Hsiao, H. Kung, and D. Vlah, "Maximizing throughput of UAV-relaying networks with the load-carry-and-deliver paradigm," in *2007 IEEE Wireless Communications and Networking Conference*, 2007: IEEE, pp. 4417-4424.

[35] O. K. Sahingoz, "Networking models in flying ad-hoc networks (FANETs): Concepts and challenges," *Journal of Intelligent & Robotic Systems,* vol. 74, no. 1-2, pp. 513-527, 2014.

[36] G. Pei, M. Gerla, and T.-W. Chen, "Fisheye state routing: A routing scheme for ad hoc wireless networks," in *2000 IEEE International Conference on Communications. ICC 2000. Global Convergence Through Communications. Conference Record*, 2000, vol. 1: IEEE, pp. 70-74.

[37] T. Clausen *et al.*, "Optimized link state routing protocol (OLSR)," 2003.

[38] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *ACM SIGCOMM computer communication review*, 1994, vol. 24, no. 4: ACM, pp. 234-244.

[39] E. M. Belding-Royer and C. E. Perkins, "Evolution and future directions of the ad hoc on-demand distance-vector routing protocol," *Ad Hoc Networks,* vol. 1, no. 1, pp. 125-150, 2003.

[40] V. Park and M. Corson, "IETF MANET Internet Draft" draft-ietf-MANET-tora-spe03. txt," ed: November, 2000.

[41] D. B. Johnson, D. A. Maltz, and J. Broch, "DSR: The dynamic source routing protocol for multi-hop wireless ad hoc networks," *Ad hoc networking,* vol. 5, no. 1, pp. 139-172, 2001.

[42] C. Perkins, S. Ratliff, and J. Dowdell, "Dynamic MANET On-demand (AODVv2) Routing draft-ietf-manet-dymo-26," *IETF, Feb,* 2013.

[43] J. Zhao and G. Cao, "VADD: Vehicle-assisted data delivery in vehicular ad hoc networks," *IEEE transactions on vehicular technology,* vol. 57, no. 3, pp. 1910-1922, 2008.

[44] I. Leontiadis and C. Mascolo, "GeOpps: Geographical opportunistic routing for vehicular networks," in *2007 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, 2007: Ieee, pp. 1-6.

[45] B. Karp and H.-T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th annual international conference on Mobile computing and networking*, 2000: ACM, pp. 243-254.

[46] V. Naumov, R. Baumann, and T. Gross, "An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces," in *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, 2006: ACM, pp. 108-119.

[47] K. C. Lee, U. Lee, and M. Gerla, "Survey of routing protocols in vehicular ad hoc networks," in *Advances in vehicular ad-hoc networks: Developments and challenges*: IGI Global, 2010, pp. 149-170.

[48] S. Schnaufer and W. Effelsberg, "Position-based unicast routing for city scenarios," in *2008 International Symposium on a World of Wireless, Mobile and Multimedia Networks*, 2008: IEEE, pp. 1-8.

[49] C. Lochert, M. Mauve, H. Füßler, and H. Hartenstein, "Geographic routing in city scenarios," *ACM SIGMOBILE mobile computing and communications review,* vol. 9, no. 1, pp. 69-72, 2005.

[50] V. Naumov and T. R. Gross, "Connectivity-aware routing (CAR) in vehicular ad-hoc networks," in *IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications*, 2007: IEEE, pp. 1919-1927.

[51] C. Lochert, H. Hartenstein, J. Tian, H. Fussler, D. Hermann, and M. Mauve, "A routing strategy for vehicular ad hoc networks in city environments," in *IEEE IV2003 Intelligent Vehicles Symposium. Proceedings (Cat. No. 03TH8683)*, 2003: IEEE, pp. 156-161.

[52] B.-C. Seet, G. Liu, B.-S. Lee, C.-H. Foh, K.-J. Wong, and K.-K. Lee, "A-STAR: A mobile ad hoc routing strategy for metropolis vehicular communications," in *International Conference on Research in Networking*, 2004: Springer, pp. 989-999.

[53] D. Forderer, "Street-Topology Based Routing (Master's thesis)," *Mannheim, Germany: University of Mannheim,* 2005.

[54] M. Jerbi, R. Meraihi, S.-M. Senouci, and Y. Ghamri-Doudane, "GyTAR: improved greedy traffic aware routing protocol for vehicular ad hoc networks in city environments," *Vehicular Ad Hoc Networks,* vol. 6, 2006.

[55] H. Füßler, H. Hartenstein, M. Mauve, W. Effelsberg, and J. Widmer, "Contention-based forwarding for street scenarios," in *1st International workshop in intelligent transportation (WIT 2004)*, 2004, no. CONF.

[56] K. C. Lee, U. Lee, and M. Gerla, "TO-GO: TOpology-assist geo-opportunistic routing in urban vehicular grids," in *2009 Sixth International Conference on Wireless On-Demand Network Systems and Services*, 2009: IEEE, pp. 11-18.

[57] P.-C. Cheng, K. C. Lee, M. Gerla, and J. Härri, "GeoDTN+ Nav: geographic DTN routing with navigator prediction for urban vehicular environments," *Mobile Networks and Applications,* vol. 15, no. 1, pp. 61-82, 2010.

[58] M. D. Dikaiakos, S. Iqbal, T. Nadeem, and L. Iftode, "VITP: an information transfer protocol for vehicular computing," in *Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, 2005: ACM, pp. 30-39.

[59] H. Stübing, M. Ceven, and S. A. Huss, "A Diffie-Hellman based privacy protocol for Car-to-X communication," in *2011 Ninth Annual International Conference on Privacy, Security and Trust*, 2011: IEEE, pp. 201-204.

[60] J. Santa, F. Pereñíguez, J. C. Cano, A. F. Skarmeta, C. T. Calafate, and P. Manzoni, "Comprehensive vehicular networking platform for V2I and V2V communications within the walkie-talkie project," *International Journal of Distributed Sensor Networks,* vol. 9, no. 7, p. 676850, 2013.

[61] X. Yang, L. Liu, N. H. Vaidya, and F. Zhao, "A vehicle-to-vehicle communication protocol for cooperative collision warning," in *The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004.*, 2004: IEEE, pp. 114-123.

[62] R. L. Lidowski, B. E. Mullins, and R. O. Baldwin, "A novel communications protocol using geographic routing for swarming uavs performing a search mission," in *2009 IEEE International Conference on Pervasive Computing and Communications*, 2009: IEEE, pp. 1-7.

[63] A. I. Alshbatat and L. Dong, "Adaptive MAC protocol for UAV communication networks using directional antennas," in *2010 International Conference on Networking, Sensing and Control (ICNSC)*, 2010: IEEE, pp. 598-603.

[64] R. Martins, P. S. Dias, E. R. Marques, J. Pinto, J. B. Sousa, and F. L. Pereira, "IMC: A communication protocol for networked vehicles and sensors," in *Oceans 2009-Europe*, 2009: IEEE, pp. 1-6.

[65] F. Yue, "Transfer of Video Images from an UAV to a Ground Station," 2013. [Online]. Available: https://www.academia.edu/5248371/Transfer_of_Video_Images_from_an_UAV_to_a_Ground_Station_Wilson_Foo_Newcastle

[66] *Riverbed's Modeler Academic Community*. (2019). [Online]. Available: https://cms-api.riverbed.com/portal/community_home.[Accessed on: 20-Feb-2016]

[67] Edimax, "Edimax BR-6288 ACL", [Online]. Available: https://www.edimax.com/edimax/global/. [Accessed on : 10-Feb-2016].

[68] B. A. Forouzan and S. C. Fegan, *TCP/IP protocol suite*. McGraw-Hill, 2006.

[69] Drone Flyer. "Drone Rules.", [Online]. Available: https://www.casa.gov.au/knowyourdrone/drone-rules. [Accessed on:10-Sept-2019].

[70] IEEE STD, "IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," *IEEE Std 802.11-1997,* pp. 1-445, 1997, doi: 10.1109/IEEESTD.1997.85951.

[71] IEEE STD 802.11a, "IEEE Standard for Telecommunications and Information Exchange Between Systems - LAN/MAN Specific Requirements - Part 11: Wireless Medium Access Control (MAC) and physical layer (PHY) specifications: High Speed Physical Layer in the 5 GHz band," *IEEE Std 802.11a-1999,* pp. 1-102, 1999, doi: 10.1109/IEEESTD.1999.90606.

[72] IEEE Std 802.11b, "IEEE Standard for Information Technology - Telecommunications and information exchange between systems - Local and Metropolitan networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher Speed Physical Layer (PHY) Extension in the 2.4 GHz band," *IEEE Std 802.11b-1999,* pp. 1-96, 1999, doi: 10.1109/IEEESTD.2000.90914.

[73] IEEE Std 802.11g, "IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Further Higher Data Rate Extension in the 2.4 GHz Band," *IEEE Std 802.11g-2003 (Amendment to IEEE Std 802.11, 1999 Edn. (Reaff 2003) as amended by IEEE Stds 802.11a-1999,*

802.11b-1999, 802.11b-1999/Cor 1-2001, and 802.11d-2001), pp. 1-104, 2003, doi: 10.1109/IEEESTD.2003.94282.

[74] IEEE Std 802.11n-2009, "IEEE Standard for Information technology-- Local and metropolitan area networks-- Specific requirements-- Part 11: Wireless LAN Medium Access Control (MAC)and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput," *IEEE Std 802.11n-2009 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, and IEEE Std 802.11w-2009),* pp. 1-565, 2009, doi: 10.1109/IEEESTD.2009.5307322.

[75] IEEE Std 802.11ac, "IEEE Standard for Information technology-- Telecommunications and information exchange between systemsLocal and metropolitan area networks-- Specific requirements--Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications--Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz," *IEEE Std 802.11ac-2013 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11aa-2012, and IEEE Std 802.11ad-2012),* pp. 1-425, 2013, doi: 10.1109/IEEESTD.2013.6687187.

[76] IEEE STD, "IEEE Standard for Information technology—Telecommunications and information exchange between systems Local and metropolitan area networks—Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012),* pp. 1-3534, 2016, doi: 10.1109/IEEESTD.2016.7786995.

[77] PCMag Australia, "The Best Drones For 2019", [Online]. Available: https://au.pcmag.com/drones/37261/the-best-drones. [Accessed on : 12-Mar-2019].

[78] Huawei, "New Video Technologies Emerging with Rollout of 2K and 4K Ultra HD Videos", [Online]. Available: https://www.huawei.com/minisite/4-5g/en/industryjsdc-f.html. [Accessed on : 16-Aug-2018].

[79] Cisco TelePresence IX5000 Series, [Online]. Available: http://www.videocentric.co.uk/wp-content/uploads/2014/11/cisco-ix5000-datasheet.pdf.

[80] R. T. Roca, "ISDN architecture," *AT&T technical journal,* vol. 65, no. 1, pp. 5-17, 1986.

[81] L. Dryburgh and J. Hewett, *Signaling System No. 7 (SS7/C7): protocol, architecture, and services*. Cisco press, 2005.

[82] CISCO. "Voice Over IP - Per Call Bandwidth Consumption", [Online]. Available: https://www.cisco.com/c/en/us/support/docs/voice/voice-quality/7934-bwidth-consume.html.

[83] R. de Silva and S. Rajasinghege, "Optimal Desired Trajectories of UAVs in Private UAV Networks," in *2018 International Conference on Advanced Technologies for Communications (ATC)*, 2018: IEEE, pp. 310-314.

[84] S. Rajasinghege and R. de Silva, "Reliability in Private Unmanned Aerial Vehicle Networks," presented at the APCEAS-0167, Sydney, 2018.

[85] Parrot for Developers, "Parrot Ground SDK", [Online]. Available: https://developer.parrot.com/. [Accessed on: 12-Feb-2019].

[86] Wireshark Wiki, "WLAN (IEEE 802.11) capture setup", [Online]. Available: https://wiki.wireshark.org/CaptureSetup/WLAN. [Accessed on : 5-Aug-2019].

[87] M. R. Silva *et al.*, "Performance Evaluation of Multi-UAV Network Applied to Scanning Rocket Impact Area," *Sensors,* vol. 19, no. 22, p. 4895, 2019.

Appendix A Storm 4 UAV

A.1 Storm 4 flying platform (V2.0)

Storm 4 Mini UAV is a lightweight UAV. It has 8-12 minutes flying time with easy to control remote controller. Storm 4 UAV is shown in Fig. A.1 and specification are given in Table A.1.

Fig. A.1 Storm 4 UAV with DEVO remote controller transmitter, Source: Storm 4 Mini UAV, http://www.helipal.com from [9]

Table A.1 Storm 4 Mini UAV

Dimension	490mm X 490 mm X 140mm
Motor to motor	450mm
Propeller size	8" X 4.5" Carbon Fibre Propeller
Take-off weight	850gm (Max safety 1350gm)
Battery	11.1V
Flight time	8-12 minutes
Max payload	250gm
Remote controller transmitter	DEVO 7 2.4GHz LCD digital transmitter

We can see the specifications of Storm 4 Mini UAV in the above table. This UAV can lift the payload of 250 grams. When we started our research, we were looking for a UAV that fits our research budget and lifts a weight of WAP. We searched across various models of UAV at that time and finalised this UAV for our research. This UAV is obsoleted in the market now, and the company launched the advanced version of this UAV. At present Storm 8 UAV are available in the market.

Appendix B: Parrot UAVs

The UAV company named Parrot manufactured different models of UAVs every year. Each UAV have their own specification and functionalities. The best part of the Parrot UAVs is its free software development kit that is available online, and a user can design his own custom application to control these UAVs. Parrot develops new UAV models every year with the new functionality and its old model are obsolete from the market from time to time. We studied and compared some models of parrot UAVs to understand the video quality difference. The specifications of these Parrot's UAVs are mentioned in Table B.1.

Table B.1 Parrot UAV model comparison with their video quality and flight time

	BeBop	**BeBop + SkyController**	**AR 1.0**	**AR 2.0**
Video resolution	1080p	1080p	480p	720p
Range	250m	2km	50m	50m
Flight time	22min	22min	12min	12min

At present, the ANAFI extended, Parrot Mambo First Person View (FPV), Parrot Mambo Code and BeBop UAV models are available on the Parrot website. We can still purchase the old version of Parrot UAVs from other distributors as they are not available on the Parrot website. Since we used the AR 2.0 UAVs for our practical testbed. Therefore, the details study of AR 2.0 UAV is given in the below section with their control, feedback and video signals.

B.1 AR 2.0 UAV

The AR 2.0 UAV is a small size commercial UAV developed by the Parrot. This UAV has the flight time of 12 minutes and control with the smartphone app. This UAV works with three types of signals (that is Control, feedback and video). The smartphone app sends control signals to the UAV in the form of control commands. The feedback signals are transmitted from UAV to the smartphone app knows as the navdata. And the live video streaming is received at the smartphone app in the form of video signals. The details of these three signals are given below.

B.2 AR UAV Control Commands

The various AR 2.0 UAV control commands are given in the AR 2.0 UAV development guide. We discuss these commands one by one and calculate how much size they occupied in the control packet.

- **Command: AT*REF=Argument 1, Argument 2, <CR>**

This control command is used for take-off, landing and emergency stop of AR UAV. It has two arguments, one is used for the sequence number, and the second one is used for controlling the UAVs. The second argument uses 32 bits to store the control signals. In this UAV control commands, the string character occupies one byte in the UDP control packet. Therefore, the command name (AT*REF=) occupied seven bytes in the UDP packet. The first argument in this command uses four bytes signed integer for the sequence number and the second argument takes the four bytes for controlling the UAV. Another one byte is used for carriage return. As such, 16 bytes are required for UDP data and 8 bytes is reserved for UDP header for this control command. We presented the UDP datagram packet for this command in Table B.2.

Table B.2 UDP packet for the control command

UDP header	**UDP data**
8 bytes	16 bytes

- **Command: AT*PCMD=Argument 1, Argument 2,.., Argument 6,<CR>**
- **AT*PCMD_MAG=Argument 1, Argument 2,..., Argument 7,<CR>**

These are two the similar control commands that are used for movement of the UAVs and have a different number of arguments. As we are interested in the maximum size of UDP packet for the control command, therefore, we consider the second command with absolute control to get the packet size. As per the earlier calculation for UDP data, we required 12 bytes for command name and one byte for the carriage return. This command has seven argument and each argument use 4-byte data. Therefore, 41 bytes are required for UDP data for this command. Table B.3 shows the UDP packet for this control command.

Table B.3 UDP packet for the control command

UDP header	UDP data
8 bytes	41 bytes

- **Command: AT*FTRIM=Argument 1, <CR>**

The AR UAV uses this control command to instruct the UAV to lie horizontally. This command has only one argument that is the sequence number. We required 9 bytes for the command name, one byte for carriage return and 4-byte data for the sequence number. Therefore, the UDP data for this command has 14 bytes, as shown in Table B.4.

Table B.4 UDP packet for the control command

UDP header	UDP data
8 bytes	14 bytes

- **Command: AT*CALIB=Argument 1, Argument 2,<CR>**

This control command is used for calibration the magnetometer of AR UAV. It has two arguments sequence number and identifier for device calibration. The UDP datagram for this command required 9 bytes for a command name, one byte for carriage return and 8 bytes for the two arguments. The UDP data for this command has 14 bytes, as shown in Table B.5.

Table B.5 UDP packet for the control command

UDP header	UDP data
8 bytes	18 bytes

- **Command: AT*CONFIG=Argument 1, Argument 2, Argument 3<CR>**
- **AT*CONFIG_IDS=Argument1, Agrument2, Argument 3, Argument 4,<CR>**

These are two similar commands used by the AR UAV and have a different number of arguments. The role of these control commands is to set a configuration option on the UAV. We consider the four arguments configuration command for packet size calculation. The UDP datagram for this command has 14 bytes for a command name, one byte for the carriage return and 16 bytes for the arguments. The total UDP data for this command will be 31 bytes, as shown in Table B.6.

Table B.6 UDP packet for the control command

UDP header	**UDP data**
8 bytes	31 bytes

B.3 AR 2.0 Navigation DataStream

Navigation data (navdata) specifies the UAV status like its altitude, camera, velocity and tag direction. This information is periodically sent to the client application, and AR 2.0 UAV uses the UDP port 5554 to send this navdata to the client application. It is in binary form and having several sections block of data called options.

B.3.1 AR UAV feedback navdata options

The details of the options that are used by AR 2.0 UAV feedback navdata are given in this section.

- **UAV battery:** float32, 4 bytes are used to store the battery information of UAV, as mentioned in Table B.7.

Table B.7 Feedback Navigation data for UAV Battery

Battery: float32	**Information**
4 bytes	0: no battery 100: Full battery

- **UAV state:** uint32, (4 bytes) is used to store the UAV state as per Table B.8.

Table B.8 Feedback Navigation data for UAV Status

UAV State: uint32	**Information**
4 bytes	0: Unknown, 1: Init, 2: Landed, 3: Flying, 4: Hovering, 5: Test 6: Taking off, 7: Goto Fix Point, 8: Landing, 9: Looping

- **UAV Compass's raw x,y,z:** int32, (4 bytes) is used to store the raw compass data as per Table B.9.

Table B.9 Feedback Navigation data for UAV Compass

UAV Compass's raw x,y,z: int32	**Information**
4 bytes	int32 magX, int32 magY, int32 magZ

- **UAV Pressure Sensor:** int32, (4 bytes), the pressure sensor is given in Table B.10.

Table B.10 Feedback Navigation data for UAV Pressure

UAV Pressure Sensor: int32	**Information**
4 bytes	For monitor the Pressure

- **UAV Temperature Sensor:** int32, (4 bytes), the temperature sensor is shown in Table B.11.

Table B.11 Feedback Navigation data for UAV temperature

UAV temperature Sensor: int32	**Information**
4 bytes	For monitor the temperature

- **UAV Wind Sensor:** float32, (4 bytes), Table B.12 shows the wind sensor.

Table B.12 Feedback Navigation data for UAV wind sensor

UAV wind Sensor: float32	**Information**
4 bytes	float32 wind_speed, float32 wind_angle float32 wind_comp_angle

- **UAV Rotation information:** float32, (4 bytes) Table B.13 presents the UAV rotation information.

Table B.13 Feedback Navigation data for UAV rotation

UAV rotation: 4 bytes	**Information**
about X-axis	Left or right tilt in degrees
about Y-axis	Forward or backward tilt in degrees
About Z-axis	Orientation in degrees

- **UAV Altitude information:** int32, (4 bytes) Altitude information is shown in Table B.14.

Table B.14 Feedback Navigation data for UAV Altitude

UAV Altitude	**Information**
4 bytes	Estimate the altitude in cm

- **UAV Linear velocity:** float32, (4 Byte) Velocity information of AR UAV is presented in Table B.15.

Table B.15 Feedback Navigation data for UAV Velocity

UAV Linear velocity	**Information**
4 bytes for each axis velocity Measure in mm/sec	Linear velocity with x-axis vx Linear velocity with y-axis vy Linear velocity with z-axis vz

- **UAV Acceleration:** float32, (4 bytes) Table B.16 shows the acceleration information.

Table B.16 Feedback Navigation data for UAV Acceleration

UAV acceleration	**Information**
4 bytes for each axis acceleration Measure in g	Acceleration with x-axis ax Acceleration with y-axis ay Acceleration with z-axis az

- **UAV motor commands:** uint8, (1 Byte) is shown in Table B.17.

Table B.17 Feedback Navigation data for UAV motor

UAV motor command	**Information**
1 byte for each motor	Four motor commands for motor 1 to 4

- **UAV tags in vision detection:** uint32, (4 bytes) is given in Table B.18.

Table B.18 Feedback Navigation data for UAV Vision

UAV vision detection	**Information**

4 bytes	uint32 tags_count, uint32 tags_type, uint32 tags_xc, uint32 tags_yc uint32 tags_width, uint32 tags_height float32 tags_orientation, float32 tags_distance

- **UAV time stamp:** float32, (4 bytes) is given in Table B.19.

Table B.19 Feedback Navigation data for the UAV timestamp

UAV timestamp	**Information**
4 bytes	UAV timestamp

B.4 Video stream

AR drone uses the proprietary video stream format that is based on H.263 UVLC (Universal Variable Length Code) format. The encoding of images is done in YCBCR colour space format, 4:2:0 type with 8 bits values. Each image is split in a group of blocks with 16-lines-hight parts of the image.

B.4.1 Initiating the video stream

In AR 2.0 UAVs, a video stream is started by sending the UDP packet on the video port as specified in Fig. B.1.

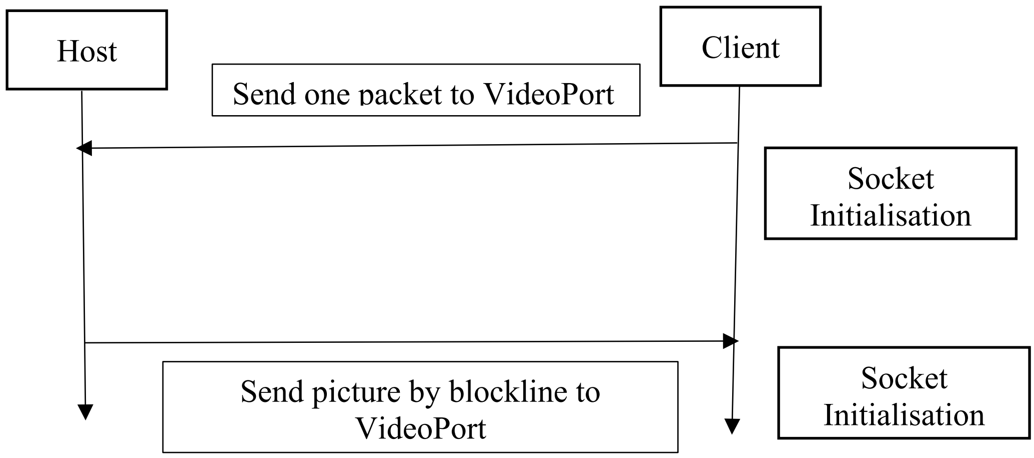


Fig. B.1 AR 2.0 Video stream initialisation

As shown in this figure, the client sends a packet to video port as a wake-up command for the host and then the host transmits a picture in the UDP packet to the client.

Appendix C: DJI UAV

C.1 DJI UAVs models

The DJI company is manufacturing a number of different UAVs each year. In the past couple of years, they developed powerful commercial UAVs. It will be interesting to know their communication services in relation to control the aircraft and to get the details of the signals that are transmitted to the ground station. We first list all the available DJI UAVs with their communication services to the ground station. This list also helps us to understand how the various DJI connect to the mobile devices as well as the wireless technology that they used to connect the aircraft.

Table C.1 Communication services for DJI UAV models

Product	**Remote Controller**	**Connectivity to Mobile Device**	**Connectivity to Aircraft**	**Supports Dual RC**
Inspire 1	Required	USB	Lightbridge	Yes
Inspire 1 Pro/Raw	Required	USB	Lightbridge	Yes
Inspire 2	Required	USB	Lightbridge	Yes
Matrice 100	Required	USB	Lightbridge	Yes
Matrice 200	Required	USB	Lightbridge	Yes
Matrice 210	Required	USB	Lightbridge	Yes
Matrice 210 RTK	Required	USB	Lightbridge	Yes

Matrice 600	Required	USB	Lightbridge	Yes
Matrice 600 Professional	Required	USB	Lightbridge	Yes
Mavic 2 Pro	Required	USB	OcuSync 2	-
Mavic 2 Zoom	Required	USB	OcuSync 2	-
Mavic 2 Enterprise	Required	USB	OcuSync 2	-
Mavic Pro	Optional	USB	OcuSync	-
Mavic Air	Optional	USB	Wi-Fi	-
Phantom 3 4K	Required	Wi-Fi	Wi-Fi, Aux	-
Phantom 3 Advanced	Required	USB	Lightbridge	-
Phantom 3 Professional	Required	USB	Lightbridge	-
Phantom 3 Standard	Required	Wi-Fi	Wi-Fi, Aux	-
Phantom 4	Required	USB	Lightbridge	-

Phantom 4 Advanced	Required	USB	Lightbridge	-
Phantom 4 Professional	Required	USB	Lightbridge	-
Phantom 4 Professional V2	Required	USB	Lightbridge	-
Spark	Optional	Wi-Fi	Wi-Fi	-

The above table shows the different UAVs models that are developed by the DJI company. These UAVs used a different type of communication technology for connectivity with aircraft and with a mobile device. Most of the UAVs used USB to connect the mobile device. But the Phantom 3 4K, Phantom 3 Standard and Spark are the only two UAVs that uses Wi-Fi technology to connect the mobile device. On the other hand, the connectivity with the aircraft is accomplished with Lightbridge, OcuSync and Wi-Fi technologies. In the following section, we discussed these AirLink technology used by the DJI UAVs.

As we can see from Table C.1, the four UAVs named Mavic Air, Phantom 3 4K, Phantom 3 Standard and Spark used the Wi-Fi technology to connect the remort controller to the aircraft. The Lightbridge technology is used by most of the UAVs, and some UAVs also use OcuSync technology.

C.2 Communication technology used in DJI UAV Models

C.2.1 Wi-Fi communication

Wi-Fi is used as the wireless communication link for both aircraft and handheld camera products. In the case of aircraft, the remote controller acts as a Wi-Fi AP and the aircraft and mobile device join it as clients. Some aircraft also acts as the AP themselves, allowing the mobile device to connect directly.

C.2.2 OcuSync

Part of the Lightbridge family, DJI's newly developed OcuSync transmission system performs far better than Wi-Fi transmission at all transmission speeds. OcuSync also uses more effective digital compression and channel transmission technologies, allowing it to transmit HD video reliably even in environments with strong radio interference. Compared to traditional analog transmission, OcuSync can transmit video at 720p and 1080p – equivalent to 4-10 times better quality, without a colour cast, static interference, flickering or other problems associated with analog transmission. Even when using the same amount of radio transmission power, OcuSync transmits further than analog at 4.1mi (7km). Before taking off, OcuSync automatically scans the environment and choose the frequency band with the lowest interference, ensuring more stable video transmission. During a flight, it sends key flight parameters back for viewing in the SDK and supports a maximum download speed of 40Mb/s for photos and videos. Additionally, since Wi-Fi uses a traditional protocol stack, it takes longer - from several seconds to tens of seconds – to get connected and to re-connect after signal loss, But OcuSync uses Cross-Layer Protocol Design, it can establish or re-establish links within one second. As well as point-to-point video transmission, OcuSync also supports wireless connections to multiple devices. For example, we can connect the DJI Goggles, remote controller, and Mavic wirelessly to OcuSync all at the same time.

C.2.3 Lightbridge

Lightbridge has been developed by DJI specifically for long-range, robust aerial communication in the 2.4 GHz band, and is used as the link between remote controller and aircraft. It provides significantly more range than Wi-Fi, with up to 5 km communication in some products. Lightbridge has 8 selectable channels. Channel selection can either be done manually or left to the radio to determine what channel has the least interference. Data rate and channel quality can be measured to understand how the channel is performing. Some remote controllers with Lightbridge wireless links also have a secondary video port. This port can be used to send the live stream in High Definition Multimedia Interface (HDMI) or Serial digital interface (SDI) format to an external device.

C.2.4 Lightbridge Accessory

DJI also has a stand-alone accessory product Lightbridge 2 that can be integrated into airframes such as the S1000. This product consists of a module that mounts on the aircraft and a remote controller for ground control. The purpose of the product is to relay remote control commands to the aircraft and relay telemetry and video data to the remote controller. The accessory provides two additional features in the live video stream compared to ready-to-fly systems using Lightbridge technology:1) Multiple video inputs on the aircraft can be combined into a picture in picture live video stream and 2) An on-screen-display (OSD) mode overlays aircraft attitude information onto the live video stream.

As mentioned earlier, DJI designed various commercial UAV models, and each of them has a different specification. The comparison of different DJI UAV models according to their flying time, their capability and functionality are presented in table C.2 and C.2.

Table C.2 Comparison of the different Phantom model of DJI

	Phantom 3 Professional	**Phantom 3 Advanced**	**Phantom 3 4K**	**Phantom 3 Standard**
Suitable For	Aerial videographers for whom high-quality UHD 4K video capabilities are a requirement	Aerial photographers who need an aircraft with an extended range and live HD view	Aerial videographers for whom high-quality UHD 4K video capabilities are a requirement	Beginners who require a ready-to-fly, affordable aerial platform with Intelligent Flight Modes
Max Transmission Distance	Up to 5 km or 3.1 miles (unobstructed, free of interferences) when it is FCC compliant; Up to 3.5 km or 2.1 miles (unobstructed, free of interferences) when it is CE	Up to 5 km or 3.1 miles (unobstructed, free of interferences) when it is FCC compliant; Up to 3.5 km or 2.1 miles (unobstructed,	FCC：1200m (outdoors and unobstructed) CE：500m	CE：500m FCC：1000m (outdoors and unobstructed)

	compliant	free of interferences) when it is CE compliant		
Video Transmission System	Built-in DJI Lightbridge Video Downlink	Built-in DJI Lightbridge video Downlink	Built-in DJI Wi-Fi video Downlink	Built-in DJI Wi-Fi video Downlink
Video Transmission Distance	Up to 5 km or 3.1 miles (unobstructed, free of interferences) when it is FCC compliant; Up to 3.5 km or 2.1 miles (unobstructed, free of interferences) when it is CE compliant	Up to 5 km or 3.1 miles (unobstructed, free of interferences) when it is FCC compliant; Up to 3.5 km or 2.1 miles (unobstructed, free of interferences) when it is CE compliant	FCC: 1200m CE: 500m (outdoors and unobstructed, flight altitude 120m)	FCC: 1000m CE: 500m (outdoors and unobstructed, flight altitude 120m)
Max FPV Preview Quality:	HD 720P @ 30fps (depending on conditions and mobile device)	HD 720P @ 30fps (depending on conditions and mobile device)	SD 480P @ 30fps	HD 720P @ 30fps (depending on conditions and mobile device)
Weight: (Including Battery And	1280 g	1280 g	1280 g	1216 g

Propellers)				
Charger And Charging Time	100 W charger for both Remote Controller and Intelligent Flight Battery included Intelligent Flight Battery charging time: 63 minutes	57 W charger for both Remote Controller and Intelligent Flight Battery included Intelligent Flight Battery charging time: 96 minutes	57W Remote and Flight Battery Charger	57 W Charger for Intelligent Flight Battery included Remote Controller is charged via USB Intelligent Flight Battery charging time: 96 minutes
Flight Time	About 23 mins	About 23 mins	About 25 mins	About 25 mins
Video Resolution	Maximum UHD 4K/30fps	Maximum 2.7K/30fps	Maximum UHD 4K/30fps	Maximum 2.7K/30fps

Table C.3 Comparison of Mavic and Spark model of DJI

	MAVIC PRO PLATINUM	**MAVIC PRO**	**SPARK**
Key Features	30-Minute Flight time Quieter Flight Foldable 4K Camera RAW Format Photos 3-Axis Mechanical Gimbal 7 km video	Foldable 4K Camera RAW Format Photos 3-Axis Mechanical Gimbal 7 km video Transmission 27-Minute Flight time	Small and Compact Quick Launch Obstacle Sensing 12 MP Camera Gesture Control Quick Shot Active Track

	Transmission		
Dimensions	Folded: 198×83×83 mm (L×W×H) Unfolded: 305×244×85 mm (L×W×H)	Folded: 198×83×83 mm (L×W×H) Unfolded: 305×244×85 mm (L×W×H)	143×143×55 mm (L×W×H)
Weight	734 g	734 g	300 g
Flight Performance	Max Flight Time: 30 minutes Max Speed: 65 kph (S - mode) 36 kph (P - mode)	Max Flight Time: 27 minutes Max Speed: 65 kph (S - mode) 36 kph (P - mode)	Max Flight Time: 16 minutes Max Speed: 50 kph (S - mode) 21 kph (P - mode)
Photography	3-Axis Mechanical Gimbal1/2.3" CMOS Max video Resolution: 4K @30 fps Max Photo Resolution: 12 MP	3-Axis Mechanical Gimbal 1/2.3" CMOS Max video Resolution: 4K @30 fps Max Photo Resolution: 12 MP	2 - Axis Mechanical Gimbal + Ultra Smooth 1/2.3" CMOS Max video Resolution: 1080 p Max Photo Resolution: 12 MP
Max Transmission Distance (Unobstructed, Free Of Interference)	Intelligent Mobile Device: 80 m Remote Controller: 2.400 - 2.483 GHz FCC: 7000 m CE: 4000 m SRRC: 4000 m	Intelligent Mobile Device: 80 m Remote Controller: 2.400 - 2.483 GHz FCC: 7000 m CE: 4000 m SRRC: 4000 m	Intelligent Mobile Device: 100 m Remote Controller: 2.412 - 2.462 GHz FCC: 2000 m CE: 500 m SRRC: 500 m

			5.745 - 5.825 GHz FCC: 2000 m CE: 300 m SRRC: 1200 m
Max Flight Time (No Wind)	30 minutes (at a consistent 25 kph)	27 minutes (at a consistent 25 kph)	16 minutes (at a consistent 20 kph)
Max Flight Distance (No Wind)	13 km	-	9 km
Operating Frequency	2.400 - 2.4835 GHz 5.725 - 5.850 GHz	2.400 - 2.4835 GHz 5.725 - 5.850 GHz	2.400 - 2.4835 GHz 5.725 - 5.825 GHz
Max Video Bitrate	60 Mbps	60 Mbps	24 bps

Appendix D: 3DR UAV

D.1 3DR Solo UAV

The 3DR Solo UAV is the most popular UAV designed by 3DR robotics. They are making more powerful UAV to use in a commercial application such as agriculture. The specification of 3DR UAV is given in Table D.1.

Table D.1 3DR SOLO UAV

Cameras: Compatible with GoPro® HERO3, 3+ and 4; optimised for HERO3+ and 4	**Max ascent speed**: 10 m/s in stabilize mode; 5 m/s in “fly” mode	**Software**: APM: Copter
Streaming video quality: 720p	**Max descent speed**: ditto	**Communication**: 3DR Link secure Wi-Fi network
Flight time: 25 minutes; 20 minutes with payload	**Max payload**: 420 g	**Propellers**: 10″ diameter 4.5″ pitch self-tightening (24 cm diameter 144 cm pitch); glass-reinforced nylon
Range:.5 miles (.8 km)	**Max altitude**: 400 ft per FAA regulation, user adjustable (122 m)	**Solo Gimbal**: Three-axis stabilization
Max speed: 55 mph (89 km/h)	**Motors**: 880 kV	Fully compatible with 3DR Solo and GoPro HERO3+ and HERO4; camera charging and stabilization only with HERO3
Frequency: 2.4 GHz	**Controller battery**: 2600 mAh	HDMI video output

	7.2 Vdc rechargeable lithium ion	
Weight: 3.3 lbs. (1.5 kg) / 3.9 lbs. (1.8 kg) with GoPro® and Solo Gimbal	**App requirements**: iOS 8.0 or later / Android 4.3 or later	Wireless software upgrade through Solo
Dimensions: 10 in. tall (25 cm), 18 in. (46 cm) motor to motor	**Solo Smart Battery**: Rechargeable lithium polymer (Lipo)5200 mAh 14.8 Vdc Weight: 1 lb. (.5 kg)	**Battery charge time**: ~1.5 hours
Flight battery: Lithium polymer 5200 mAh 14.8 Vdc	**Autopilot**: Pixhawk 2	

D.2 3DR Solo Drones Architecture

The 3DR drones work with Solo system. The Solo is a Linux based system that is directly connected to Pixhawk autopilot. The Pixhawk is an advanced autopilot system that is designed and manufactured by 3DR robotics. The main functionality of the Pixhawk system is controlling the UAVs and perform recovery in return to launch events. The communication protocols used by the Pixhawk system is MAVLink telemetry protocol. This protocol provides communication between the Pixhawk system and onboard Linux computer as well as to downstream such as a controller or mobile phone Solo app.

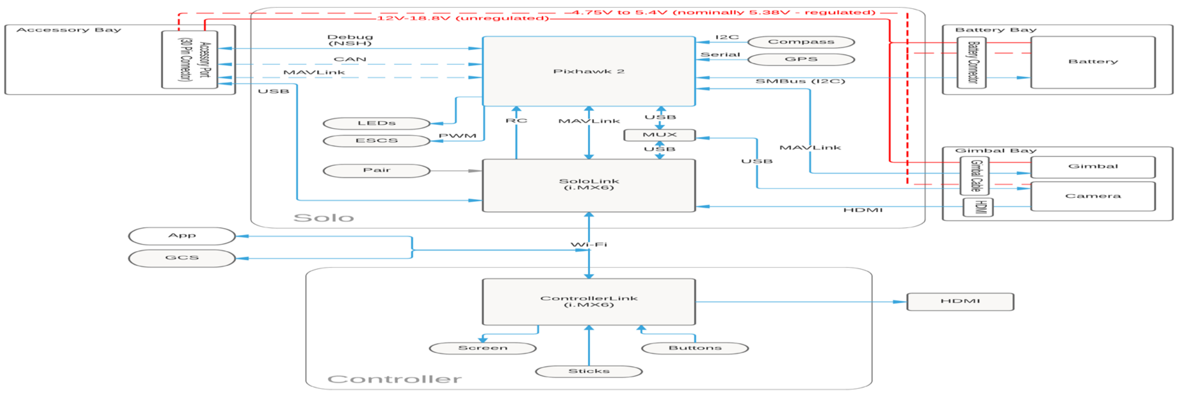


Fig. D.1 Solo system architecture, Source: Solo system, https://3drobotics.github.io/solodevguide/concept-architecture.html

Fig. D.1 shows the Solo system architecture that is designed for the 3DR UAV. In this Fig., the solid blue line indicates the data flow, and its arrow shows the direction of data flow, the red line is a power line and dashed-blue data line specifies the payload bays. Solo system of 3DR UAV is responsible for various services such as communication proxying, Received Signal Strength indication (RSSI) testing, video encoding and communication with Solo app. In the normal Solo operation, these services are always on and they are restarted after a crash or shut down.

D.3 Video transmission

The transmission of video from 3DR drone to ground controller device or mobile device is accomplished with various steps. The videos are captured by GoPro camera mounted on 3DR UAV. The GoPro is connected with UAV's Solo system with HDMI cable. The video is first encoded by HDMI encoder and iMX6. iMX6 used h.264 or with the gstreamer encoding. The encoded video is transmitted to the controller with the Wi-Fi connection. The controller has h.264 video decoder for decoding the video. After that, it is forward to HDMI output and UDP relay to phone. This video is received to Solo app using the Wi-Fi connection. The video pipeline for video transmission is shown in Fig. D.2 below.

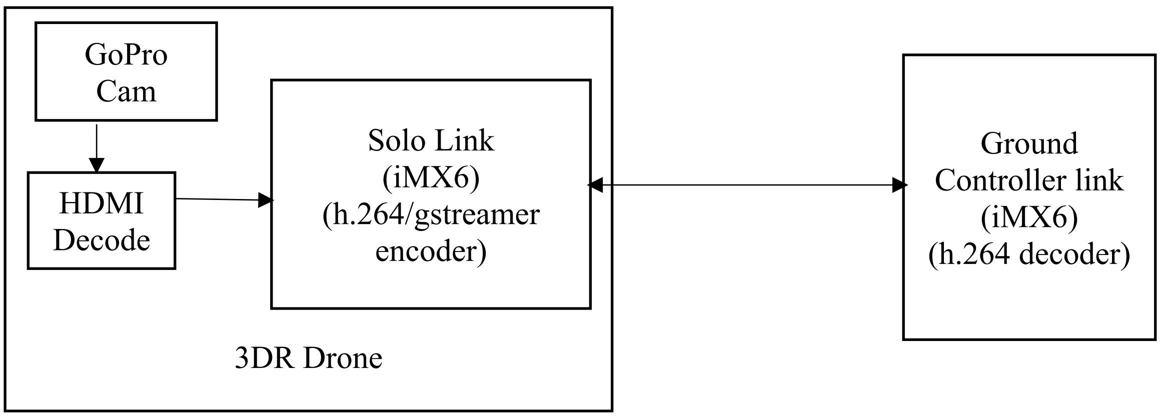


Fig. D.2 3DR UAV video pipeline

Appendix E: Edimax WAP

E.1 Edimax BR-6288ACL

In our practical experiment, we used Edimax Br-6288 ACL WAP shown in Fig. E.1. The functionality of this WAP is mentioned in Table E.1.

Fig. E.1 Edimax Br-6288ACL WAP, Source: BR-6288ACL, https://www.edimax.com [67]

Table E.1 Edimax WAP

Functions	•Supports router, access point, range extender, Wi-Fi Bridge and WISP modes •Guest network •Up to 10 SSIDs (2.4GHz x 5 and 5GHz x 5) with VLAN support in AP mode •IGMP proxy and IGMP snooping •DDNS and DHCP •Port triggering for special applications •Virtual server and demilitarized zone (DMZ) hosting •MAC/IP filter and URL blocking •iQoS for smart bandwidth management •Static routing •UPnP architecture

	•VPN pass through (IPSec/PPTP) •Wi-Fi schedule control •Fault tolerance firmware upgrade
Hardware Interface	•1 x RJ-45 10/100M WAN/LAN combo port (*An Edimax RJ-45 Splitter included.) •1 x micro USB power port •Status LED indicators •Internal high gain antenna •WPS/Reset button •Wireless Normal/Green mode switch
Management & Installation	•Multi-language user interface •Supports remote management •System status and security log •Firmware upgradable •Smart iQ Setup, no CD required •Supports App smart setup (iOS 7 or Android 4 and above are required for smartphone or tablet setup)
WAN	•Supports WISP connection mode •Supports RJ-45 cable/xDSL modems •WAN protocol: PPPoE, static IP, dynamic IP, PPTP and L2TP
Output Power & Sensitivity Gain (5GHz)	Output Power •11a(54Mbps): 13±1.5dBm •11n(20MHz, MCS7): 12±1.5dBm •11n(40MHz, MCS7): 12±1.5dBm •11ac(80MHz, MCS9): 11±1.5dBm Receive Sensitivity •11a(54Mbps): -69±2dBm •11n(20MHz, MCS7): -68±2dBm

	•11n(40MHz, MCS7): -64±2dBm •11ac(80MHz, MCS9): -57±2dBm
Output Power & Sensitivity Gain (2.4GHz)	Output Power •11b (11Mbps): 14±1.5dBm •11g (54Mbps): 13±1.5dBm •11n (20MHz, MCS7): 12±1.5dBm •11n (40MHz, MCS7): 12±1.5dBm Receive Sensitivity •11b(11Mbps): -83±2dBm •11g (54Mbps): -69±2dBm •11n(20MHz, MCS7): -67±2dBm •11n(40MHz, MCS7): -64±2dBm
Security	•64/128-bit WEP, WPA, and WPA2 security •QoS for critical operations •SPI anti-DoS firewall
Memory	•4MB NOR Flash •64MB RAM
Humidity & Temperature	•10-90% (non-condensing) •0-40ºC
Power Adapter	DC 5V, 1.2A
Dimensions	•215(H)mm x 70(W)mm x 70(D)mm
Certifications	•FCC/CE