

SIGNALING PROTOCOLS FOR LOCAL AREA NETWORKS OF DRONES

Prabhu Jyot Singh and Rohan de Silva

School of Engineering and Technology CQ University Sydney, Australia

ABSTRACT

Unmanned Aerial Vehicles (UAVs) or drones have become extremely popular and are used in various commercial applications. When multiple UAVs communicate and work together, they form a UAV network. A private UAV network or Local Area Network of Drones (LoDs) is a special type of UAV network which has the minimum number of UAVs to carry out a certain task. All UAVs in a LoD use the wireless medium to send and receive the data as well as the control signals. An organization or a single owner will be more interested in this type of network, where they want multiple UAVs to scan an area, communicate with each other, and send all the images and live video streams to a single ground station. The currently available commercial UAVs can send the video signals to and receive control signals only from their own ground station controllers. However, in an LoD 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 and performance.

In this paper, we study the frame formats of existing control, feedback, and data messages of commercial AR UAVs and propose a new approach to construct the payloads of control and feedback frames that are suitable for an LoDs. We compare the performance of our approach of single control and feedback frame for all UAVs in a LoD branch with that of separate control and feedback frames for each UAV. We calculate and compare the UAV node processing delay in both types of signaling mechanisms and show that the single control and feedback frame signaling has less delay on the average.

KEYWORDS

UAV communication, Private UAV Network, LANs of Drones, Signaling Frames, Control and feedback signaling

1. INTRODUCTION

Commercial UAVs provide a variety of services in various domains such as agriculture, surveillance, rescue, and construction. UAVs have become a powerful tool for commercial applications with their live video recording and streaming capabilities. The live video streaming helps the owner of a UAV to manage a large area while staying near the ground station or even at home. For example, a person may use a UAV to monitor an agricultural farm and receive live streaming videos of the farmland at a ground station [1].

In commercial applications of UAVs, the UAVs can provide more flexibility to their owners if they work together in a network. We can use several UAVs and form an LoD to be used in some of these commercial applications [2]. The communication mechanism is a key issue in a LoD. The communication in an LoD happens from one UAV to its neighbor UAV in tandem and finally, to the ground station and vice-versa.

Most modern commercial UAVs provide the facility to communicate between UAV and a smartphone app, in which the smartphone first sets up a Wi-Fi connection with the UAV and then control signals are transferred from the smartphone app to the UAV in packet format. The UAV, in turn, sends the feedback signals and streaming video to the smartphone. The feedback signal has information about UAV's flight status, battery status, flight altitude and many others. Thus, signaling protocols play a key role in UAV to smartphone communication. In a LoD, there is only one ground station for all UAVs which can be a computer or a smartphone that acts as the controller as well as the video receiver. Sending separate control signals to each UAV or receiving separate feedback signals from each one of them is inefficient and would lead to a 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 UAVs movements as the UAVs and the ground station will not receive the control and feedback signals on time. Therefore, the control and feedback signal packets used for flying a single UAV has to be studied and should be adapted appropriately to be used in LoD networks.

The structure of the rest of the paper is as follows. Following a brief discussion of LoD topology in section 2, we discuss the signaling protocols used in two popular types of commercial UAVs. In section 4, we introduce the new signaling packets for LoD networks. Section 5 shows via simulations the performance improvement of LoD networks using this new approach. Finally, the paper is concluded in section 6.

2. REVIEW OF LODS TOPOLOGY

As mentioned in the above section, an LoD network is a network with several UAVs. An LoD has a star-connected relay topology, where the ground station is at the headend and all UAVs are located in the branches [3]. A simple LoD is shown in Figure 1. Each UAV in this network has the capability to communicate with its neighbor UAVs in the same branch. The distance between two UAVs in any two different branches of an LoD is larger than the maximum Wi-Fi coverage distance of 100m, except in the case where the branches are made to be closer to improve reliability [4].

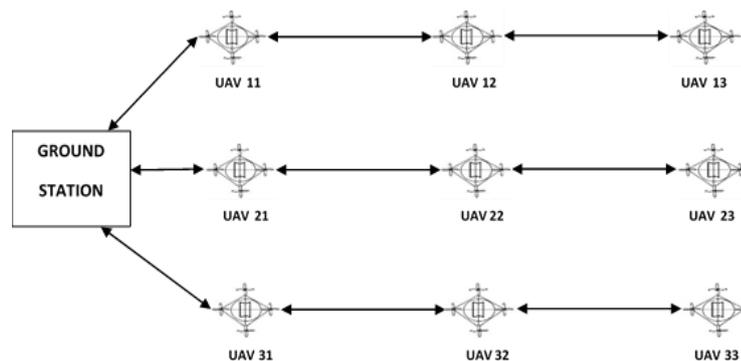


Figure 1. A LoD with three branches

The control, feedback, and data messages in an LoD branch follow the same path as shown by bidirectional arrows in Figure 1. In an LoD, each UAV has a unique Internet Protocol (IP) address as well as a Media Access Control (MAC) address. With reference to the LoD network in Figure 1, let us assume that there are m branches and each branch has n UAVs. The first UAV of each branch is directly connected with the ground station through the wireless connection. In

the next section, we discuss the control and feedback signaling of existing commercial UAVs manufactured by two popular companies.

3. SIGNALING PROTOCOLS IN EXISTING COMMERCIAL UAVS

At present, there are two established companies, DJI [5] and Parrot [6] 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 minimize 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 Figure 2. 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.

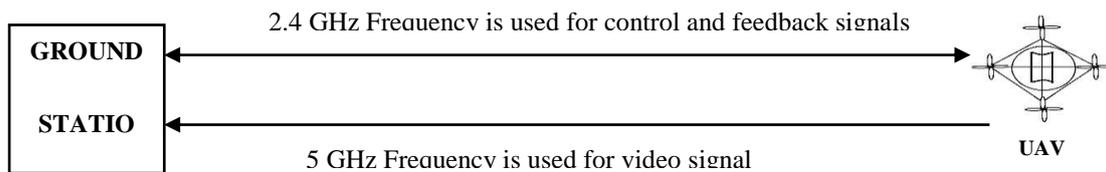


Figure 2. Signalling in 2.4 GHz

Some UAVs swap these two frequency channels to transmit their control and data signals. In this type of design, the 5 GHz frequency is used to send control and feedback signals, and the data is transmitted over the 2.4 GHz frequency. DJI has some UAV models (e.g. Phantom 2 Vision and Vision +) that use this convention. Figure 3 shows this type of communication arrangement. Both types of communication systems have their own advantages as well as disadvantages. In this paper, we discuss the signaling method used in AR 2.0 drones and modify it to design the signaling for LoD networks.

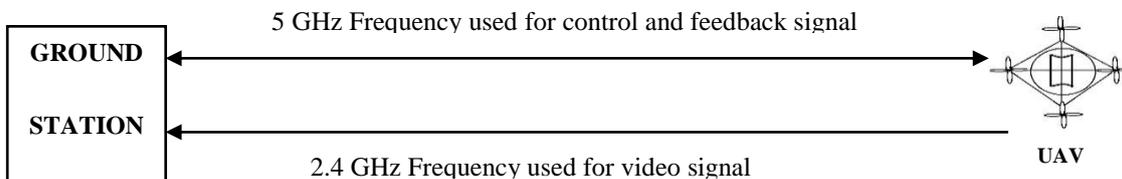


Figure 3. Signalling in 5 GHz

3.1. Parrot's AR UAVs

As described in their development guide, AR 2.0 UAVs use various control signals to represent different control commands. These control signals are transmitted from the smartphone to the UAV on Wi-Fi. AR 2.0 UAVs have four types of communication services, first to control the UAV, second for the navigation data, third for the video stream, and the last for the control port information [7].

- **Controlling and configuration:** AR 2.0 UAVs are controlled by sending the AT commands on the User Datagram Protocol (UDP) port 5556. These commands are sent regularly 30 times per second to control the UAVs but at least within 2 seconds to prevent the Wi-Fi connection being lost from the UAV. A single UDP packet can carry one or more of these commands.
- **Information about the UAV navigation data (navdata):** This is the second type of communication service in AR 2.0 UAVs. Once, the controller connection is established with the UAV and the control signal is transferred from the smartphone to the UAV, UAV sends its navdata to the smartphone. The navdata has information such as the status, speed, engine rotation speed and the battery level of the UAV. In the design of AR 2.0 UAVs, the UDP port number 5554 is allocated to receive this navdata information. This information is sent to the client 15 times per second in demo mode and 200 times per second in full mode.
- **The video stream and Control Port:** AR 2.0 UAVs send their video data in UDP or Transmission Control Protocol (TCP) packets, depending on the version. The UDP port 5555 is assigned to receive the video packets. AR 2.0 UAVs use TCP protocol and AR 1.0 UAVs use UDP protocol for the transmission of the video stream. The TCP port 5559 is used to send critical data. It is used to retrieve the configuration information data.

3.2. AR 2.0 UAV Control Packet

As we discussed in the previous section, once a controller is bounded with a UAV, it sends control signals continuously to the onboard UAV system. These control signals are carried in UDP packets as we saw earlier in section 2. We first study the UDP packet of AR 2.0 UAV and then, consider how it can be adapted for signaling in LoDs when a common control packet and a common feedback packet is used for all UAVs in a branch.

AR 2.0 UAVs use various control commands to send the control signals to the onboard UAV system. These control commands are application data that is generated by the controller (a mobile app or a computer program). These application data are passed to the transport layer and a UDP control packet datagram is created. A UDP datagram has two parts, an 8-byte header and a 1024 bytes payload. The UDP datagram of an AR 2.0 UAV is presented in Table 1. The AR 2.0 development guide provides a list of mostly used control commands for AR 2.0 UAVs.

Table 1. AR 2.0 UAV UDP datagram

UDP header	UDP data (control commands)
8 bytes	1024 byte

In AR 2.0 UAV control commands, each control command starts with three characters AT* (i.e. three 8-bit words with values 41(16), 54(16), 2A (16)) followed by a command name, an equal sign, a sequence number, and an optional list of comma-separated arguments whose meaning depends on the command. A UDP packet of AR 2.0 UAV can hold one or more control commands, separated by newlines (byte value 0A (16)). But an AT command must reside in a single UDP packet and splitting an AT command in two or more UDP packets is not possible. In AR 2.0 UAVs, strings are encoded as 8-bit ASCII characters with a carriage return character

(byte value 0D (16)) represented by <CR>, hereafter called a newline delimiter. We studied all the AR 2.0 UAV control commands to find the maximum UDP packet size. AR 2.0 UAV uses similar types of control command with a different number of arguments, for e.g., the two commands AT*PCMD and AT*PCMD_MAG. As we are interested in the maximum size of UDP payload for a single control command, we consider the command with maximum argument size from among the similar types of commands. Table 2 shows the details of the UDP data for each control command.

Table 2. UDP data for control commands

Command	UDP Data (in bytes)			Total UDP data
	for command name	for arguments	carriage return	
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

We can see from Table 2, the maximum size of the payload required for the UDP data for one control command of a 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. Wireshark traces showed there is a maximum of three control commands in a single UDP packet. Even if all these control commands are sent together, we require only 130 bytes of payload, and therefore, they can fit into a single UDP packet. These control commands are transmitted in a frame from the ground station to the AR 2.0 UAV. Wireshark showed that a control frame has 8 bytes of UDP header, 20 bytes of IP header and 14 bytes of Ethernet header. We investigated and found that the Wireshark Wi-Fi frame capturing does not work well with Windows and it gives incorrect Ethernet header details [8]. 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. We use standard Wi-Fi 802.11 MAC frame format for the AR 2.0 UAV control frame. Table 3 shows the maximum size of the control frame for AR 2.0 UAV which has 192 bytes. The ground station usually sends 30 control commands per second to the UAV as we verified using Wireshark.

Table 3. Control Frame for a LoD 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

3.3. AR 2.0 UAV feedback packet

Table 4 provides information about the navdata that is transmitted from the AR 2.0 UAV to the ground station. UDP port 5554 is used to send the navdata from the UAV to the ground controller. This navigation data is received periodically (less than 5ms) by the client application and it has various information such as flight speed, altitude, distance, roll, pitch, camera, and much more. Each UAV sends this information in a single UDP packet. As shown in Table 4,

standard navdata has 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 options 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. These fields are used to specify what type of navigation data is received by the ground station. The information in options 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 feedback frame also contains 4 bytes of a checksum at the end of the frame.

Table 4. AR 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.

Table 5. 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

3.4. AR 2.0 UAV Video packet

AR 2.0 UAV uses a TCP packet for video transmission. The Wireshark video frame also shows an incorrect Ethernet header size. Therefore, we use a standard MAC header of 802.11 to construct the video frame shown in Table 8 which has 1534 bytes.

Table 6. Video Frame

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

4. MANAGEMENT OF CONTROL AND FEEDBACK FRAMES IN LODS

In an LoD network, we can treat each UAV individually and simply send the control messages from the ground station to each one of them and receive separate feedback and data frames from each of them 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. Therefore, in section 4.3.1, we first investigate this situation and find the maximum number of UAVs in a branch that we can have to send control information and receive navdata using a single UDP packet.

4.1. Single Control and Feedback frame per each branch

In this section, we discuss the single control and feedback frame mechanism to manage the signaling in a LoD. We know that the Maximum Transfer Unit (MTU) in a Wi-Fi network is 2304 bytes [9]. 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 a 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 did not consider the checksum in the feedback data, then also we can have a maximum of 4 UAVs. This arrangement is shown in Figure 4.

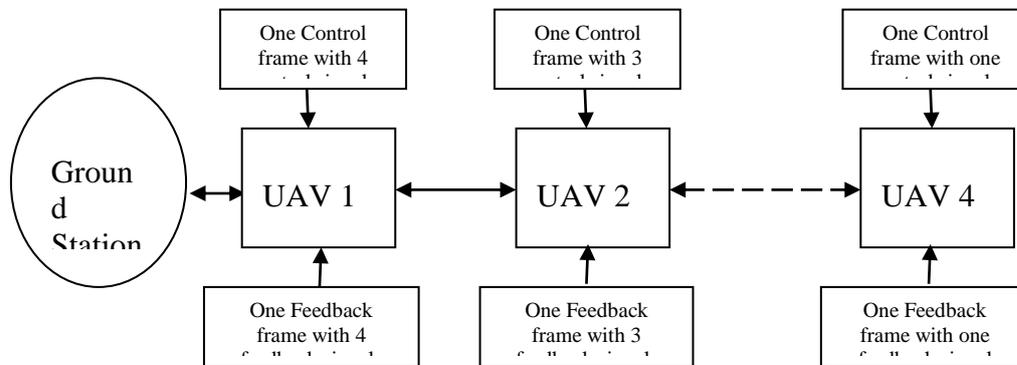


Figure 4. Single control and feedback frames for UAVs in one branch of a LoD

Table 7. The single control frame format in a LoD

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

In a LoD network, the ground station can create a single control frame for 4 UAVs in a branch. This frame has a payload of $130 \times 4 = 520$ bytes. This single frame also has 8 bytes UDP header, 20 bytes IP header, 30 bytes 802.11 MAC header and 4 bytes FCS as shown in Table 7. UAV 1 will receive a single frame of 582 bytes. UAV 1 processes this frame, extracts its control command and repacks the remaining data into a new frame before sending it to UAV 2. Now, UAV 2 decapsulates the frame to extract its control command. This process repeats at each UAV and continues until the last UAV (UAV 4) receives its control command as shown in Figure 4.

In a Similarly manner, each UAV processes the feedback frame traveling in the opposite direction from UAV 4 to the ground station. The ground station receives a feedback frame with feedback data of 4 UAVs as shown in Table 8.

Table 8. The single feedback frame format in a LoD

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

In this LoD network, UAV 4 creates a feedback frame of 558 bytes and sends it to UAV 3. UAV 3 processes this incoming feedback frame and reconstruct a new single frame by adding its feedback data and passes it to UAV 2. The process continues until up to UAV 1 node. Finally, the ground station receives a single feedback frame that has the feedback information of all 4 UAVs. Table 9 shows the control and feedback frame processing at each UAV along a branch.

Table 9. Frame processing at each UAV node in a 4 UAVs LoD network.

Node	Receiving Control Frame Size	Leaving Control Frame Size	Receiving 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

We used only four UAVs in a branch in this LoD network with this new signaling mechanism. However, in an LoD network with more than four UAVs in a branch, these single control, and feedback frame signaling mechanisms 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 these feedback frame has the feedback information of four UAVs.

4.2. Separate Control and Feedback frame per each node

The second method of signaling is the transmission of separate control and feedback frame for each UAV as shown in Figure 5. In order to compare with the case of a single control frame, we consider an LoD with only 4 UAV nodes in a branch. The ground station generates four control frames, one control frame for each UAV and broadcasts them. Assuming that only UAV 1 is in its Wi-Fi range, which is the case in a LoD, 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 remaining three control frames are broadcast. Then, they will be picked up by UAV 2 and placed in its queue. The process continues until the 4th 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 generate a feedback frame and broadcasts it. As shown in Figure 6, UAV 4 sends its own feedback frame to UAV 3. UAV 3 sends its own feedback frame and that of UAV 4 to UAV 3 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 the same queue is used to store the incoming control and feedback frames but a separate queue is used for video frames.

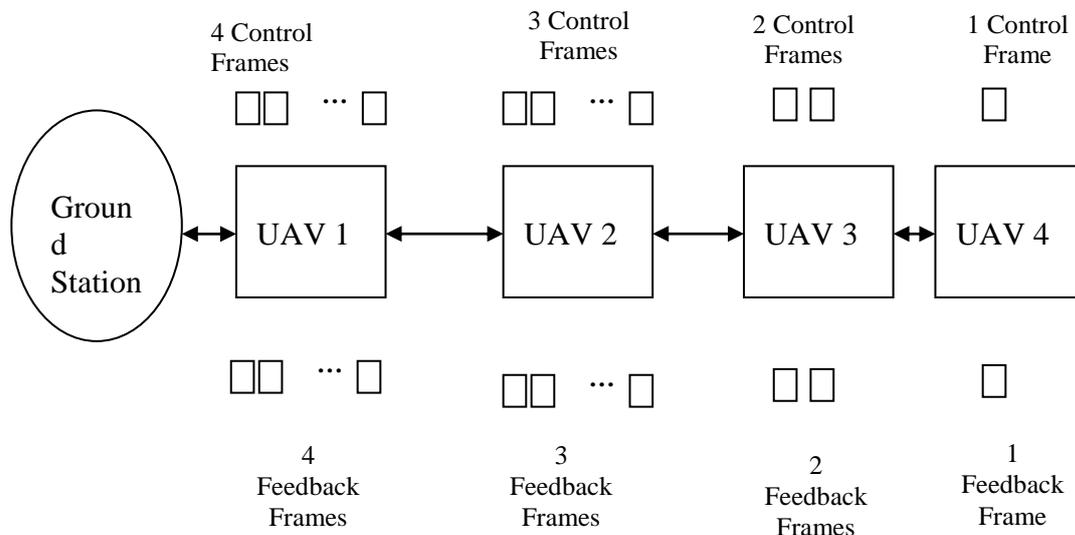


Figure 5. Separate control and feedback frames for 13 UAVs in one branch of a LoD

5. ANALYSIS OF THE PROPOSED SIGNALING MECHANISMS FOR LODS

We discussed two mechanisms in section 4 to manage the signaling in an LoD. In this section, we will compare the single control and feedback packet signaling mechanism with the separate control and feedback signaling mechanism. We first calculate the total delay from the ground station to each UAV node for both scenarios in the LoD and then compare these delays. 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 (single control and feedback packet), 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 UAV will always be less than 100 m, 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 (R). 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 purposes, the total delay for both signaling mechanisms will be taken as the sum of PD and QD.

5.1. The total delay for separate control and feedback frame

In this section, we calculate the total delay for separate control and feedback frame signaling for the LoD network. To calculate the delay, we have to first find the arrival rate (λ) of control and feedback frames at each UAV and also the service rate (μ). The service rate is 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 (i.e. control, feedback, and video). Therefore, we have to consider the time shared by the processor to process control, feedback and video frames. AR 2.0 UAV development guide does not provide the information about the processor time sharing to process these three frames. But we have already calculated the size of control and feedback frames and we also know as to how many of these control and feedback frames can be send in one second to and from the AR 2.0 UAV. Hence, we can find the average processing speed to process these frames used by AR 2.0 processor. In AR 2.0 UAV, for smooth communication with the ground station, a control frame should be sent in a regular basis, usually, 30 times per second. It means AR 2.0 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 ($0.192 \times 8 \times 30$) kbps. 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 feedback frame is 66.96 ($0.558 \times 8 \times 15$) kbps. We need the 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 rate of raw video data rate. We captured many video frames in Wireshark and calculated the average video processing speed. We used the Wi-Fi frame header size for calculating the processing speed of video frames. 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.

Since in an LoD network, each UAV has to communicate with its neighbors, we used Edimax WAP [10] on each UAV to enable communication. 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 LoD network. Therefore, in the LoD 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.

In delay calculation, we used queuing theory formulas. The control and feedback delay calculation for the separate control and feedback signaling is given in Tables 10 and 11, respectively. We used the following parameters to calculate the 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
- L_q is the mean number of frames in the queue
- QD is the queuing delay
- PD is the processing delay
- TD is the total delay
- 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

➤ **Control Frame Delay with 4 UAVs in a LoD branch**

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

$$TD = QD + PD \tag{1}$$

QD is calculated with the following formula

$$QD = L_q / \lambda \tag{2}$$

$$L_q = (\rho^2 / (1 - \rho)) \tag{3}$$

$$\rho = (\lambda / \mu) \tag{4}$$

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

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

$$PD = (\lambda / \mu) \tag{6}$$

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

$$TD = (((\lambda / \mu)^2 / (1 - (\lambda / \mu))) / \lambda) + (\lambda / \mu) \tag{7}$$

Table 10. Total delay for control frame with separate control and feedback frame

UAV node	Arrival no. of a control frame	The arrival rate of control frames (192 B *30) (λ) (kbps),	TD (seconds) Where $\mu = 1.43$ 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

➤ **Feedback Frame Delay with 4 UAVs network**

Table 11 shows the queuing and processing delay calculation for a feedback frame for the separate control and feedback signaling in the LoD network. We found the number of feedback frames arriving at each UAV and calculated the arrival rate of feedback frames.

Table 11. Total delay for feedback frame with separate control and feedback frame

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

We found the total delay at each UAV node in the LoD network with the help of Tables 10 and 11. The total delay calculation is shown in Table 12. We also calculated the delay from the ground station to each UAV node as shown in Table 12.

Table 12. The delay from the ground station to each UAV node with separate control and feedback frames

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

5.2. Cross-Layer Design of Single control and Feedback Frame

We used a cross-layer design to process each incoming frame. In cross-layer design, the frame is processed at the MAC layer without going through higher layers in the TCP/IP model. 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 f frames per second, respectively. We used the same queuing delay formulas to calculate the delay of frames with a single control and feedback frame for all 4 UAVs. The delay calculations for control and a feedback frame are given in Tables 13 and 14. We also calculated the delay from the ground station to each UAV node as given in Table 15.

➤ Control frame processing of a LoD branch with 4 UAVs

Table 13. Total delay for the single control frame

UAV Node	The arrival rate of control frame per second (λ) (kbps)	TD (seconds) where $\mu = 1.43$ Mbps
1	139.680	0.097678397
2	108.480	0.075860197
3	77.280	0.054041998
4	46.080	0.0322238

➤ Feedback frame processing of 4 UAVs

Table 14. Total delay for the single feedback frame

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

Table 15. The delay from the ground station to each node with a single control and feedback frame

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

5.3. Delay comparison

We calculated the delay from each UAV node to the ground station for both types of signaling mechanisms in the last section. Table 12 shows the delay for the current signaling mechanism,

where separate feedback and control frame is used for each UAV and the delay for the single control and feedback signaling mechanism in Table 15. The delay comparison graph shown in Figure 6 was drawn with the help of Table 12 and 15. We can see from the graph that our newly developed single control and feedback frame mechanism has a lower delay compared to the existing signaling method. The reduction in delay as a percentage with the new signaling mechanism is presented in Table 16.

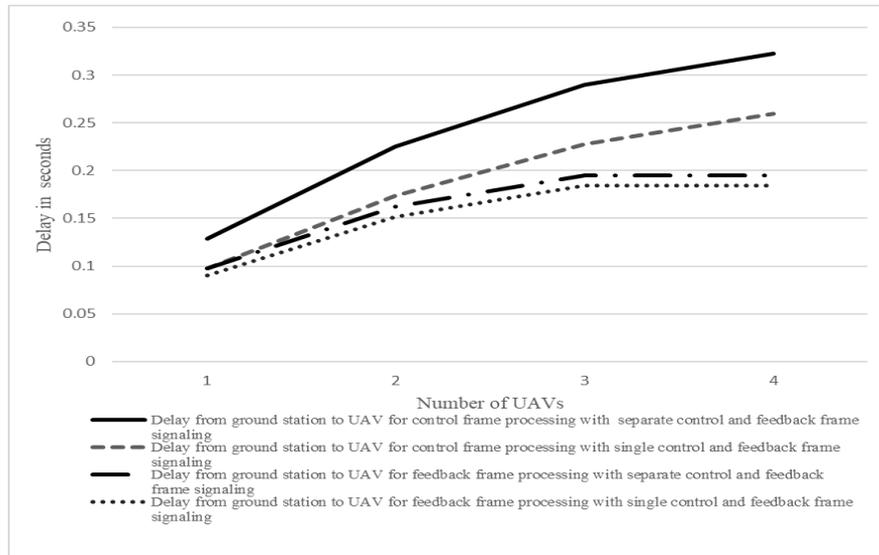


Figure 6. Delay comparison between single control and feedback frame for all 4 UAVs and separate control and feedback frames for each UAV

Table 16. Reduced Delay in percentage for control and feedback frame in 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

The delay in both signaling mechanisms is due to the high volume of video frame processing by each UAV. We can see from Table 16 that the new signaling mechanism reduces the delay around twenty percent for control frame processing and at least five percent for feedback frame processing. Thus, the single control and feedback frame signaling mechanism is a better approach for the signalling in LoDs as far as the delay is concerned.

In an LoD 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 [4]. The research result in [4] 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. [11] have conducted multiple experiments to evaluate the performance of the 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 a newly proposed signaling mechanism. Hence, the frame loss rate with separate control and feedback signaling for each UAV is given by 0.18% ($0.9/500 \times 100$). 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 a cost of increasing the frame loss rate.

6. CONCLUSION

The main purpose of this paper is to investigate the frame rates of the current signaling mechanism in LoDs and proposing a new signaling mechanism to reduce the delay from the ground station to each UAV. For this purpose, we studied the control, feedback, and video frame rates of AR 2.0 UAVs that are transmitted between a UAV and ground station. In the LoD 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 LoDs 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 an LoD 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 an LoD network.

REFERENCES

- [1] A. Nixon, "Best Drones for Agriculture 2018: The Ultimate Buyer's Guide", [Online]. <https://bestdroneforthejob.com/drone-buying-guides/agriculture-drone-buyers-guide/>, July 2018.
- [2] R. de Silva, <https://www.leisureinformatics.com/t31-private-uav-networks-and-lans-of-drones#33>
- [3] R. de Silva and S. Rajasinghege, "Optimal Desired Trajectories of UAVs in Private UAV Networks", International Conference on Advanced Technologies for Communications (ATC), Ho Chi Minh City, 2018, pp. 310-314.
- [4] S. Rajasinghege and R. de Silva, "Reliability in Private Unmanned Aerial Vehicle Networks", APCEAS-0167, Sydney, 2018, pp. 428-434.
- [5] DJI, [Online] <https://www.dji.com/au>
- [6] Parrot, [Online] <https://www.parrot.com>
- [7] AR 2.0 Development guide, [online], <https://jpchanson.github.io/ARdrone/ParrotDevGuide.pdf>
- [8] WLAN (IEEE 802.11) capture setup, [online], <https://wiki.wireshark.org/CaptureSetup/WLAN>
- [9] 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," in IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012), vol., no., pp.1-3534, 14 Dec. 2016
- [10] Edimax router, [online] https://www.edimax.com/edimax/merchandise/merchandise_detail/data/edimax/global/wireless_routers_ac600/br-6288acl/
- [11] 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.

AUTHORS

Prabhu Jyot Singh was born in Bikaner, India, in 1985. He received the B.E. degree in Computer Science & Engineering from the University of Rajasthan, Rajasthan, India, in 2007, and the M.S. degree in Advanced Computer Science from the University of Leicester, Leicester, U.K, in 2010. In 2012, he joined the Department of Computer Science & Engineering at Manda Institute of Technology, Bikaner, as an Assistant professor. At present, he is a PhD candidate in Business and Informatics at the Department of Engineering & Technology, CQUniversity, Sydney Campus. His current research interests include communication protocols for UAVs. He undertakes his PhD research under the guidance of Dr Rohan de Silva who is his principal supervisor. Dr Indra Seher acts as his associate supervisor.



Dr Rohan de Silva is a lecturer of the School of Engineering and Technology of the CQUniversity Australia. He has specialised in computer networks and lectures ICT courses. Rohan has obtained a PhD and a MEng degree in Electrical and Telecommunications Engineering from the University of New South Wales (UNSW), Australia and a BSc Eng (Hons) degree from the University of Moratuwa, Sri Lanka. Rohan is a senior member of the Institute of Electrical and Electronics Engineers (IEEE) and a member of the Association for Computing Machinery (ACM). He is also a life member of the Association of Medical Doctors Asia (AMDA) Sri Lankan Chapter. Rohan has published widely in the areas of data communications as well as the application of statistics and data mining. He has been a paper reviewer, an international program committee member and a session chair of a number of IASTED, WSEAS and IEEE conferences. He was also an organising committee member of the IEEE International Conference on Networking (ICON) 2003 held in Sydney. Rohan invented private UAV Networks or LANs of Drones in 2014 that can be used for agriculture & fisheries, natural disaster and surveillance applications and pioneered the UAV communications research at CQ University Sydney Campus.

