International Journal of Advancements in Electronics and Electrical Engineering – IJAEEE

Volume 8 : Issue 1 - [ISSN : 2319-7498] - Publication Date: 27 Dec, 2019

Building Multi–UAV Relay Communication System using Robot Operating System

Arushi Maheshwari and Hwang-Cheng Wang

***Abstract*— The need for multi-UAV systems has been increasing with the extensive use of UAVs not only in the military activities but also in various civilian activities like search and rescue, agriculture, traffic management, building inspection, recreation, etc. But multi-UAV systems have their limitations when it comes to increased distance and the presence of obstacles in the path because of the range limitations of communication modules. Most of the communication modules like telemetry modules 3DR Sik2, Xbee, Wi-Fi modules have a communication range of up to a few kilometers, leading to less efficient multi - UAV systems in terms of long flight distance and obstacles in the path. In this paper, a relay communication system using Robot Operating System (ROS) is proposed whose first stage has been implemented and the second stage is undergoing. In this system, one of the UAVs acts as a relaying node between the UAV far in distance and GCS when the distance is large or if there are obstacles in the path. The system is implemented using two UAVs but can be implemented using more UAVs or relaying nodes with some changes in the system.**

***Keywords*—Relay system, ROS, multi - UAV, communication modules, GCS**

I. Introduction

The industry for UAV systems has flourished since the past two decades due to their wide range of applications in various fields. The UAV industry has been moving towards multi - UAV systems in order to increase the efficiency of various operations such as military surveillance, disaster management, fire control, pollution control, etc. Some of the advantages of multi-UAV systems include the ability to provide a wide coverage area, a robust system, interconnectivity between vehicles even in the presence of obstacles, etc.

Many of the complex missions require to fly over large distances with communication between the vehicles remaining unhindered by the obstacles in between. A UAV equipped with the available communication modules such as telemetry radio and, Bluetooth all work up to a certain range and gets interrupted because of obstacles like high mountains, trees, etc. For various missions such as search and rescue, surveillance, monitoring, the drawbacks of communication systems in UAVs can lead to failure of these missions. Therefore, the implementation of relay nodes disguised as UAVs can ensure interconnectivity between the UAVs and the ground control station (GCS) through relaying and routing the data sent by any of the UAVs or GCS.

In this paper, we have implemented a multi-UAV relay system suited for a number of applications with traffic inspection and control being one of them. The system is implemented using the Robot Operating System, with dependencies on MAVROS and Mavlink. The system is implemented using two UAVs and one ground control station but can be extended up to more than two UAVs. The two UAVs and the GCS are implemented as ROS nodes which exchange commands such as take-off, land, go-to, and altitude by publishing and subscribing over the ROS topics. The system is illustrated in Figure 1.

The UAVs are equipped with onboard computers to provide them with a platform for ROS. In this work, onboard computers used are Raspberry Pi Model 3 B with a 64-bit processor and the speed of CPU is 1.2 GHz. The Raspberry Pi is connected with telemetry radio on one USB port to communicate with either GCS or UAV and Pixhawk on the other USB port t**38** | P a g e o send flight commands to drive the UAV.

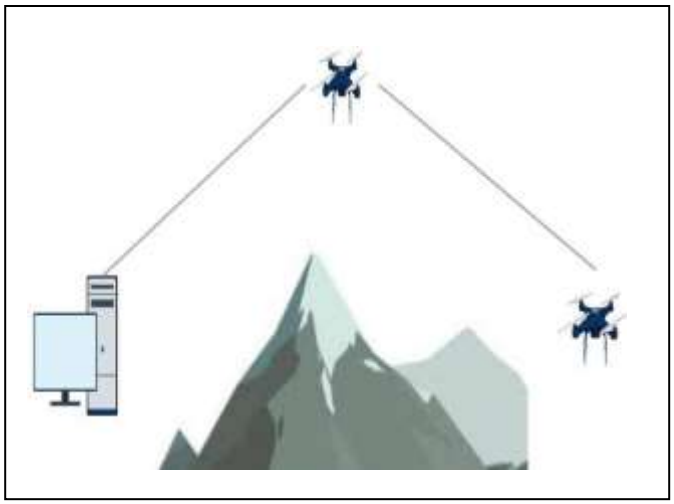

Figure 1. Relay communication system

II. Related Works

Many researchers have been working on different aspects of multiple UAV relay systems and working towards improving the performance and efficiency of such systems. The paper [2] focused on building the network architecture for multi-UAV systems. The paper described various networking architectures and data requirements for different links and nodes that can be implemented in multi-UAV systems. The paper [3] developed a more efficient system to provide power line inspection up to long distances using a relay communication system. The relaying node sends images or videos to GCS for monitoring the path of UAVs and to far-away stations for detailed analysis of power lines.

The authors of [4] discussed the various challenges faced by a swarm UAV system, such as networking and security challenges, and collision problems and also presented key challenges in the civilian applications of multi - UAV systems.

International Journal of Advancements in Electronics and Electrical Engineering – IJAEEE
Copyright © Institute of Research Engineers and Doctors
Volume 8 : Issue 1 - [ISSN : 2319-7498] - Publication Date: 27 Dec, 2019

The paper [6] focused on improving the algorithm for relay communication system by controlling the UAV heading angle. The authors of [7] developed a relay communication network between Autonomous Underwater Vehicle (AUV) and GCS with UAV being the relaying node. With the help of the UAV, the AUV sends all the data to the GCS and vice versa. In [8], the authors developed an artificial potential field path planning, providing a system to keep the communication intact between dynamical path changing vehicles and also supporting collision detection. They tested the system with the help of MATLAB simulation.

III. Hardware Setup

The section presents the hardware structure of the multi – UAV relay system. The system consists of two UAVs with one of the UAVs as a relay node (also referred to as UAV “Parent”) and the other as UAV “Child”. The relay node is connected to GCS and UAV “Child” in the presence of an obstacle or larger distance and is responsible for receiving and routing data from one node to the other.

The system consists of the following major hardware blocks responsible for the proper functioning of the mission:

1. **UAV:** The aircraft is equipped with structural components, a flight controller, an onboard computer, power system, and a battery. The UAVs used in this implementation are built from commercially available UAV Pixhawk kits. The UAV type chosen for the work is the quadcopter type.
2. **Pixhawk - Flight Controller:** The flight controllers used in the flight are Holybro Pixhawk 4 which provides 32 - bit ARM processor with 2 MB memory and 512 KB RAM. The flight controller is equipped with various sensors to get the position and altitude of the UAVs. The onboard sensors that the Pixhawk 4 is equipped with include an accelerometer, magnetometer, and barometer. Besides, it also has a GPS module attached to itself.
3. **Onboard computer:** The onboard computer used in this project is Raspberry Pi Model 3B -running Ubuntu Mate operating system and providing support to the Robot Operating System. The onboard computer is responsible for the following functions:
    a. To exchange data between with UAV or GCS,
    b. To receive or send command and decode the instructions from high-level language to machine-level language,
    c. To provide message routing and error checking.
4. **Communication Module:** The communication module used is Holybro 3DR Telemetry Radio Module (915 MHz). It will establish communication between the UAV and also between the two UAVs and will provide a medium to transmit information from one end to another.
5. **GCS Software:** The ground control software is built on the basis of the repository in [9] which runs on the Windows platform. It is an interface for issuing commands to the nodes and showing the positions of the UAVs to the user.

The GCS will be a Windows-based system, connected with telemetry radio to the relaying node “Parent” and to the “Child” node in the initial stage. It will be running the GCS Software mention in [9]. The relaying node “Parent” will be equipped with two Telemetry radio modules, one connected to the GCS and the other connected to the “Child” node. Both the “Parent” and “Child” will be running the multi - UAV relay communication program on the Robot Operating System. The “Child” node will be equipped with one telemetry radio, connected to GCS initially and then to the relaying node “Parent”.

IV. Building the multi - UAV relay communication program

A. *Robot Operating System*

The system has been programmed using the Robot Operating System (ROS), MAVROS, Mavlink. The Robot Operating System consists of the following basic components:

1. **Node:** A node is an executable that uses ROS to communicate with other nodes. In this system, both the UAVs and GCS act as nodes with UAV “Parent” being the relaying node.
2. **Topics:** A topic is a stream of messages with a defined data type either built-in or user-defined. The nodes exchange data or communicate with each other by publishing messages to a topic to send messages as well as subscribing to a topic to receive messages.
3. **Messages:** ROS data types used when subscribing or publishing to a topic. The data type can be built-in ROS data types or user-defined. The ROS messages used in this system are:
    a. **Data Frame message:** This contains the source ID, the destination ID, length of the payload and the payload itself containing the commands.
    b. **Channel Status message:** This checks the status of the channel by keeping a track of messages sent and received successfully.

The three basic structures of ROS are depicted in Figure 2.

International Journal of Advancements in Electronics and Electrical Engineering – IJAEEE
Copyright © Institute of Research Engineers and Doctors
Volume 8 : Issue 1 - [ISSN : 2319-7498] - Publication Date: 27 Dec, 2019

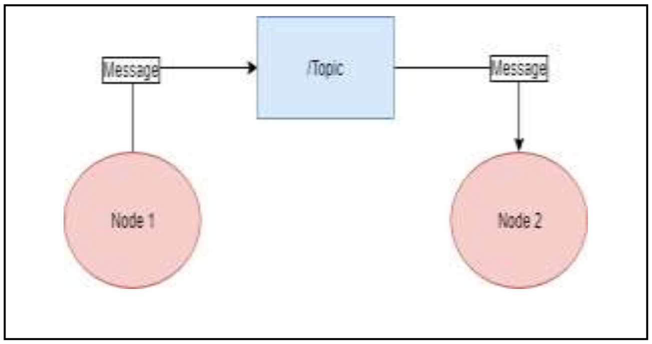

Figure 2. ROS architecture 1

B. *Functions of onboard computer*

The multi-UAV relay communication program will start when the Raspberry Pi will power up.
Upon powering up, the program will take care of the following functions:

1. To control and monitor the flight path of UAV and report it to GCS on a regular basis.
2. To receive message routed to it and convert it from high-level language to machine language.
3. To receive messages which are not targeted to it and route those to the correct destination ID.

There are two interfaces attached to the onboard computer. One interface is for the communication module, to set up a link between the two nodes in order to transmit data, and the other one is for connection with the Pixhawk flight controller, in order to send commands to take off, land, etc.

C. *The relay communication approach*

To receive, retransmit, save packets, nodes use the source id and destination id field defined in the data frame.

1. **For relaying node "Parent"**
   a. If the packet has the destination ID of Parent node itself, then it will save the packet and process it to decipher the command or data it contains.
   b. If the packet has the destination ID of Child or GCS, it will receive and retransmit the packet to the respective id.
2. **For terminal node "Child"**
   a. In the initial stage, it will receive commands from the GCS node and perform the necessary action.
   b. In the relay stage, the Child only accepts packets from the Parent node and deciphers the data it contains and acts accordingly.

A flow chart describing the relay communication approach at both Child and Parent nodes is illustrated in Figure 3 and Figure 4 respectively.

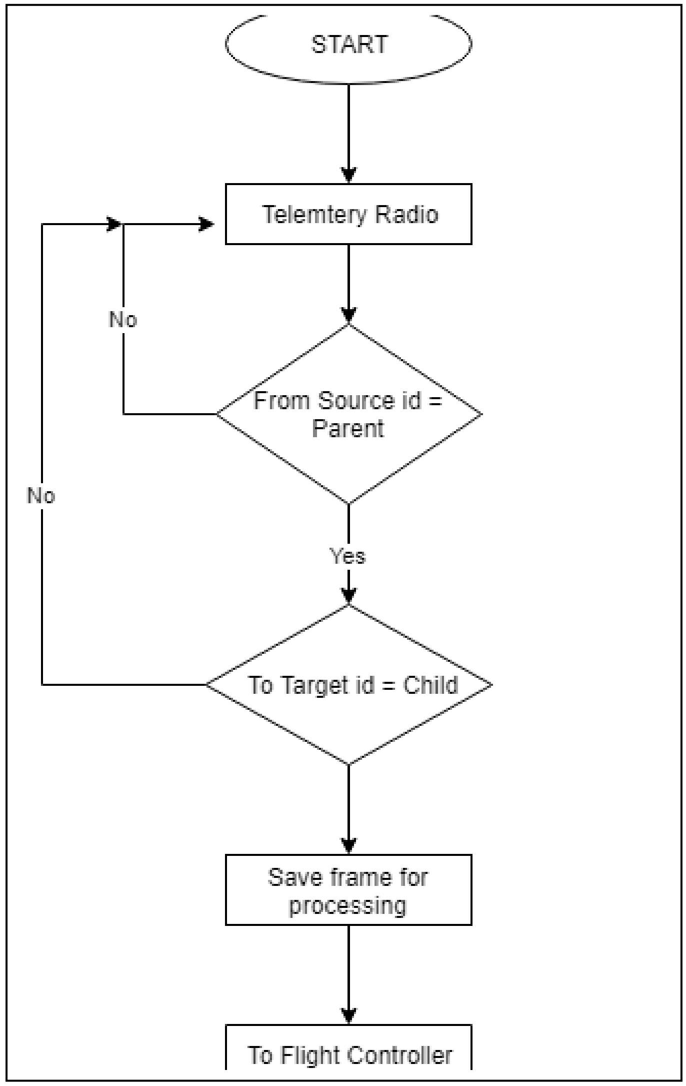

Figure 3. Relay communication approach for Child node

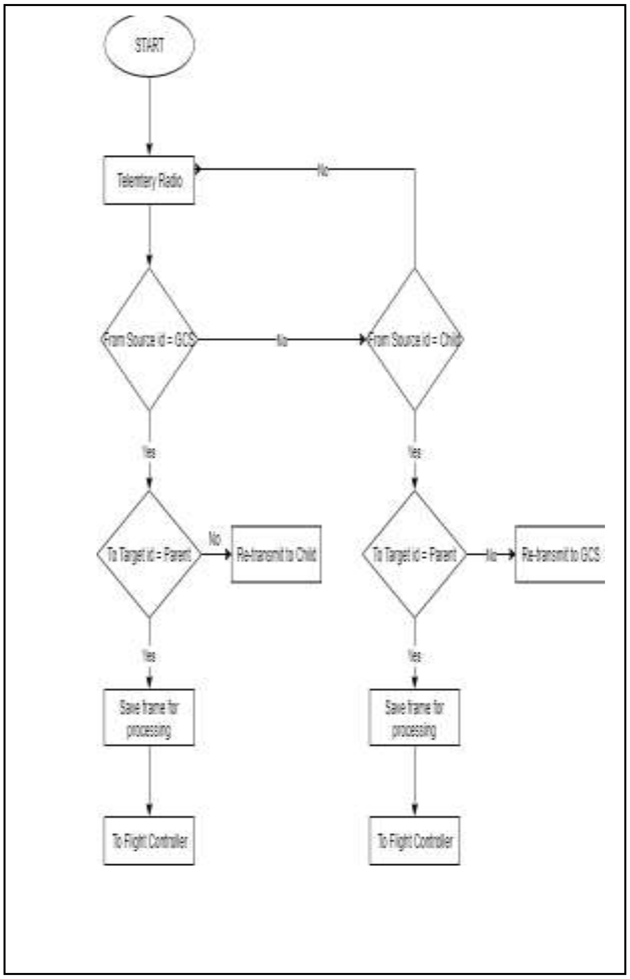

Figure 4. Relay communication approach for Parent node

V. Calculations

To measure the response rate and efficiency of the system, the system can calculate the time take by a packet to reach the targeted node from the GCS via the relaying node.

To achieve this, a time stamp was added to the packet at the GCS (t_gcs) indicating the time it left the node. Then, at the relaying node, it adds to more fields to the packet: the time at which it reached the relaying node (t_rr) and the time at which the relaying node routed the packet (t_lr). When this packet reaches the targeted node, it calculates the time taken by the packet to reach the targeted node from GCS by subtracting the time stamp added by GCS from the current time (t_now). Also, it can calculate, if needed, the time taken by the packet to reach relaying node from GCS in a similar way.

Time taken to reach targeted node = t_now – t_gcs (1)

Time taken to reach relay node = t_rr – t_gcs (2)

VI. Testing

This section contains the result of the program running after the whole setup is connected and booted up.

This project is still under work. The first stage of the paper, i.e., building the multi - UAV relay communication system on the ROS, has been completed and tested successfully. Now, the second stage of the project, i.e., conducting a field test with UAVs for the distance range is underway.

On the startup process, the onboard computers on the UAVs start the ROS program for relay communication. The system first checks if the number of ports connected to telemetry radio and Pixhawk is enough and then it verifies the connection to Pixhawk and to Parent/Child/GCS depending on the node. Then, it starts ROSCORE, the command to start ROS and then eventually MAVROS. After this, all the nodes get connected to each other. The screenshot of the boot program running on the relaying node "Parent" is given below in Figure 5.

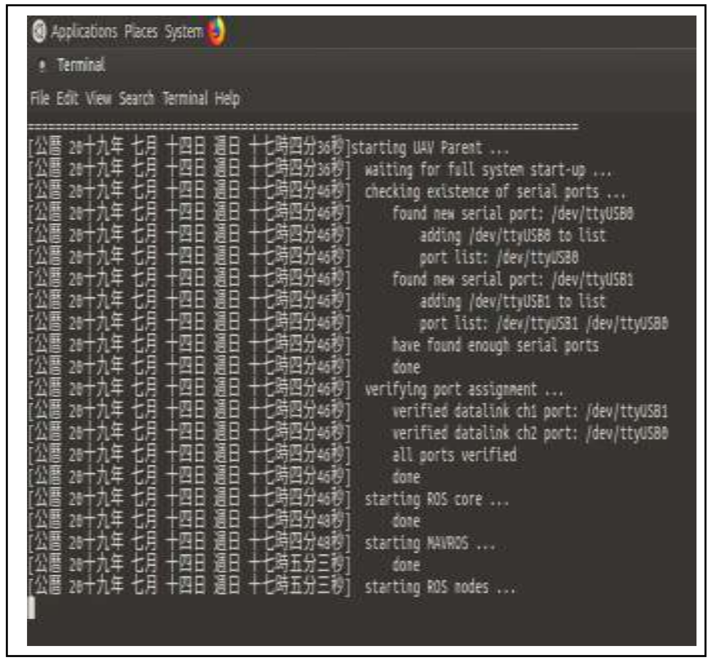

Figure 5. Screenshot of UAV node Parent 1

VII. Conclusion and Future Work

This paper provides an insight towards the implementation of multi - UAV relay communication system using two UAVs and one GCS. In today's world, the mentioned system can be employed to monitor a large area of land or traffic - reducing cost, manpower, and also increasing safety. The implementation is still under-going and improvements are being made in regular intervals. This implementation involves two UAVs equipped with flight controllers and onboard computers which are connected to each other through radio communication modules. The GCS is running a software implemented by [9] which sends commands to the relay node and is equipped with a display to monitor the flight path.

In the future, an application of such a system can be implemented by deploying more than two UAVs in the relay system. One of the applications that the paper is focusing on in the future is its implementation for a traffic control mission. More than two UAVs in a relay system can be deployed in a large area to monitor the traffic in that respective area. The

UAV can also issue a speeding ticket to the vehicles violating the rules directly on to their phones. All the data captured by the UAVs can go to Traffic Police's cloud server through relaying nodes or directly through the node capturing data. In the case of Multiple UAVs in the system, an algorithm of choosing or routing the data through the best path of relaying can be implemented too.

Acknowledgment

The research was partially supported by the TEEP@Asia Plus Program. Lindsay, Lee, Emerson, Jerry, and Gary are some of the students who were of great help during the implementation of the first stage of the system.

References

[1] Li B, Jiang Y, Sun J, Cai L, Wen CY. Development and Testing of a Two-UAV Communication Relay System. *Sensors (Basel)*. 2016; 16(10):1696. Published 2016 Oct 13. doi: 10.3390/s16101696 [Link]

[2] (19) (PDF) Communication and Networking of UAV-Based Systems: Classification and Associated Architectures. Available from: https://www.researchgate.net/publication/314098042_Communication_and_Networking_of_UAV-Based_Systems_Classification_and_Associated_Architectures [accessed Jul 12 2019] [Link]

[3] Deng, C.; Wang, S.; Huang, Z.; Tan, Z.; Liu, J. Unmanned aerial vehicles for power line inspection: A cooperative way in platforms and communications. J. Commun.2014, 9, 687–692. [Link]

[4] Shakhatreh, H.; Sawalmeh, A.H.; Al-Fuqaha, A.; Dou, Z.; Almaita, E.; Khalil, I.; Othman, N.S.; Khreishah, A.;Guizani, M. Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges.IEEE Access2019,7, 48572–48634.[Link]

[5] Lee, W.; Lee, J.Y.; Lee, J.; Kim, K.; Yoo, S.; Park, S.; Kim, H. Ground Control System Based Routing for Reliable and Efficient Multi-Drone Control System. Appl. Sci. 2018, 8, 2027. [Link]

[6] P. Zhan, K. Yu and A. L. Swindlehurst, "Wireless Relay Communications with Unmanned Aerial Vehicles: Performance and Optimization," in IEEE Transactions on Aerospace and Electronic Systems, vol. 47, no. 3, pp. 2068-2085, July 2011.[Link]

[7] T. A. Johansen, A. Zolich, T. Hansen and A. J. Sørensen, "Unmanned aerial vehicle as communication relay for autonomous underwater vehicle — Field tests," 2014 IEEE Globecom Workshops (GC Wkshps), Austin, TX, 2014, pp. 1469-1474. [Link]

[8] Cetin, Omer and Ibrahim Zagli. "Continuous Airborne Communication Relay Approach Using Unmanned Aerial Vehicles." Journal of Intelligent & Robotic Systems 65 (2012): 549-562. [Link]

[9] https://github.com/HKPolyU-UAV/Mini_GCS

About Author (s):

I believe the most interesting point is how using Robot Operating System can help us program UAVs/robots without taking into consideration the underlying communication protocols and network architecture. This makes the work of a programmer much easier.

Multi-UAV systems represent the future of UAVs in myriad of practical applications. This work presents the implementation of a prototype of such systems. Details are provided in hope of helping and inspiring subsequent research.