IET Research Journals

Research Article

IET Journals
The Institution of Engineering and Technology

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

The Design and Implementation of Aerial Communication Using Directional Antennas: Learning Control in Unknown Communication Environments

Songwei Li[1†], Chenyuan He[1†], Mushuang Liu[1†], Yan Wan[1], Yixin Gu[1], Junfei Xie[2], Shengli Fu[3], Kejie Lu[4]*

[1] *Department of Electrical Engineering, University of Texas at Arlington, Arlington, TX 76019, USA*
[2] *Department of Computing Science, Texas A&M University-Corpus Christi, Corpus Christi, TX 78412, USA*
[3] *Department of Electrical Engineering, University of North Texas, Denton, TX 76207, USA*
[4] *Department of Computer Science and Engineering, University of Puerto Rico at Mayagüez, PR 00681-9000, Puerto Rico*
† *The first three authors contributed equally to this paper.*
* *E-mail: yan.wan@uta.edu*

Abstract: To support an increasing number of commercial multi-UAV applications, robust UAV-to-UAV communication is critical. A promising solution is aerial communication using directional antennas (ACDA), with features like long communication distance with low power consumption, broad bandwidth, and interference rejection. Nevertheless, ACDA requires the automatic alignment of directional antennas, which is not easy to achieve considering the imperfect communication environment unknown in advance and the limited sensing devices onboard due to the constrained payloads and power sources. In this paper, we design and implement a new ACDA system, including the platform, communication, computing, control, middleware, and interface components. Practical implementation issues for the emergency response application are also considered. The ACDA system features a communication and control co-design, where the communication quality indicator, received signal strength indicator (RSSI), serves as the goal function for antenna alignment. The solution also features a reinforcement learning (RL)-based directional antenna control algorithm that learns the unknown communication environment models. The performance of the ACDA system is verified using simulation studies, field tests, and disaster drills.

1 Introduction

Unmanned Aerial Vehicle (UAV) networking extends the capability of a single UAV to multiple UAVs, and is foreseen to have broad uses in a wide spectrum of commercial UAV applications, such as on-demand emergency [1, 2], surveillance [3], reconnaissance [4], and remote sensing [5]. Several communication schemes for low-altitude UAV platforms have been developed, such as omni-directional networking [6, 7] and cellular-based communication [8, 9].

For applications such as emergency response, it is crucial to develop broadband and long-distance on-demand communication capabilities, such that monitoring videos at remote locations can be transmitted over a long distance to emergency management centers in real time. Such a communication architecture should not rely on ground infrastructure support, and hence can be applied whenever needed in scenarios where infrastructures (e.g., cellular networks) are not available [1, 2]. Compared to the omni-directional communication solutions, the use of directional antennas allows the energy to focus along a certain direction, and thus offers various benefits such as extended communication distance, reduced antenna power consumption, and reduced communication interference. Per our knowledge, studies on directional antennas and narrow-beam communication schemes have been focused on air-ground communication, but not for UAV-to-UAV communication [10, 11].

In the past few years, we developed several solutions using UAV-carried directional antennas to realize the on-demand broadband and long-distance UAV-to-UAV communication (see Figure 1), called the aerial communication using directional antennas (ACDA) system [1, 2, 12–14]. When the ground devices are of non-line of sight (NLOS), the UAVs dispatched to the air serve as relays to establish communication between the ground devices and solve the NLOS challenge. Paper [12] verified the feasibility of long-distance and broadband UAV communication through directional antennas using experimental studies, and developed an early version of the ACDA system where the UAV locations are fixed. In order to enable a robust communication channel subjec to UAV mobility, the autonomous alignment of antenna directions is needed. We thus developed an autonomous alignment solution that includes two UAV-to-UAV channels [13, 14]. The directional antenna-equipped channel transmits application-oriented data such as video streams, and the other low-rate omni-directional antenna-equipped channel transmits the control and command signals, such as GPS locations of the remote UAV and synchronization signals. In paper [13], a proportional-integral-derivative (PID) controller was designed to align antenna headings under wind disturbances based on the GPS signals transmitted through the omni-directional channel. In paper [14], a complete ACDA system was designed and implemented, with detailed descriptions of the UAV platform configuration, onboard autopilot, communication system, and antenna heading control. For the antenna heading control, a Linear Quadratic Gaussian (LQG) controller was developed to minimize a quadratic GPS-based tracking error. An initial scan algorithm that achieves automatic antenna direction alignment without the presence of GPS signals was also developed. The performance of the ACDA system was verified using both simulation studies and field tests.

In order to improve the UAV-to-UAV communication quality, communication quality indicators can be utilized to assist with the antenna controller design. Widely used wireless communication quality indicators include the channel throughput [15], bit error rate (BER) [16], signal-to-noise ratio (SNR) [17], and received signal strength indicator (RSSI) [18, 19]. As studied in [15–17], all of these indicators are related to the transmitting and receiving signal powers. Our special interest here is the RSSI, which measures the received signal power that is transmitted from the transmitting

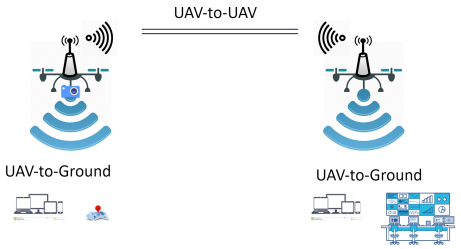


Fig. 1: Illustration of the broadband long-distance communication infrastructure using controllable UAV-carried directional antennas.

antenna, and can be directly obtained from the received signal at the receiving antenna without additional measurement equipment. With the same transmitting signal power, a stronger RSSI leads to a higher SNR, and in turn a lower BER [16] and a larger channel capacity [15]. As such, we adopt the RSSI as the communication performance indicator to quantify the performance of UAV-to-UAV wireless communication channel in ACDA.

In [20], the RSSI was used as an additional measurement signal to compensate GPS signals for the automatic alignment of directional antennas, and a proportional-integral (PI) controller was used for antenna control. This controller is designed based on two assumptions: 1) the RSSI model is known ahead of time, and 2) the strongest RSSI is achieved when two directional antennas point towards each other. These assumptions may not hold in an imperfect communication environment, considering the existence of reflection, refraction and absorption by buildings, obstacles, and interference sources, which complicate the communication channel model. In this paper, we address this issue and develop an autonomous antenna heading alignment solution that optimizes the antennas' headings to achieve maximal RSSI in real imperfect environments, with the communication channel model unknown in advance. In particular, we develop a Reinforcement Learning (RL)-based online optimal controller that learns the environment-specific communication channel model in real time.

The contributions of this paper are summarized as follows.

- *Unified Communication Channel for Application, Control and Command Data.* Application, control, and command data share the same directional antenna-equipped UAV-to-UAV channel. This networking design is very different from the previous versions which separate the high-rate directional antenna-equipped application channel and the low-rate omni-directional antenna-equipped control and command channel. This solution improves the robustness of the ACDA system, as the omni-directional antenna-equipped control and command channel is sensitive to interferences in long-distance set-ups.
- *RL-based Directional Antenna Control that Maximizes Communication Performance in Unknown Communication Environments.* The solution features a communication and control co-design, where the communication quality indicator, received signal strength indicator, serves as the antenna control's goal function. A RL-based online optimal controller is developed to find the optimal heading angles that maximize the RSSI of directional antennas in an imperfect environment. This solution is also coupled with an algorithm that learns an environment-specific communication channel model in an unknown communication environment. This antenna control solution maximizes the communication performance in real settings, as it does not require a known and perfect communication channel model as was assumed in the previous versions of ACDA.
- *Integrated Design and Implementation of a High-performance ACDA System.* We develop an ACDA system with an integrated design and implementation of the communication, control, and computing components. The implementation adopts the Robot Operating System (ROS) to facilitate communication and data transmissions. The computing functionality is implemented using BeagleBone Black, a compact low-power single-board computer suitable for UAV applications [21]. Compared to Arduino adopted in our previous ACDA systems, BeagleBone Black features a powerful CPU, multiprocessing capability, and ROS support.
- *Application to Emergency Communication Provision with the Consideration of Practical Needs.* We apply the ACDA system in emergency scenarios, by providing an on-demand broadband communication channel to transmit both optical and infrared monitoring videos of the disaster zone over a long distance to the emergency management center [1, 2]. Special considerations of emergency practices are considered in the design and implementation of the ACDA system. In particular, we develop an interface using ROS and Qt, a cross-platform application framework [22]. The interface displays at ground stations the status information of the entire system, such as GPS, antenna headings, and RSSI for system diagnosis. It also displays video streams captured by the remote UAV. In addition, a water-resistant enclosure is designed to protect the ACDA system from adverse environmental conditions for practical use.

The rest of this paper is organized as follows. In Section 2, we describe the ACDA system design, including both hardware and software structures. In Section 3, we elaborate the RL-based on-line directional antenna control solution. Section 4 verifies the directional antenna control solution and evaluates the performance of the ACDA system using simulation studies, field tests and disaster drills. Section 5 concludes this paper.

2 The ACDA System Design

In this section, we first provide an overview of the ACDA system, and then provide the detailed description and analysis for each component, including the UAV platform, communication, computing, antenna heading control, cameras, ROS, interface, and water-resistant design.

2.1 Overview

The objective of the ACDA system is to provide a robust and cost-effective long-distance and broadband UAV-to-UAV communication channel for on-demand emergency communication, and meanwhile, to offer a user-friendly interface that simplifies the operation and diagnosis procedures. The ACDA system is composed of a pair of integrated subsystems on both the local side and the remote side. The subsystem on each side includes a laptop, a UAV platform, and the onboard communication, computing, and control components (see Figure 2).

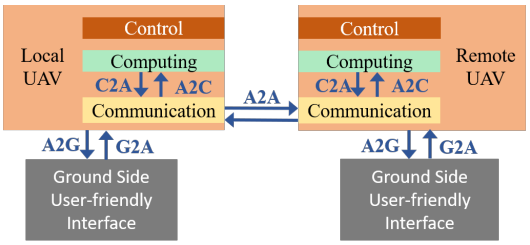


Fig. 2: Overview of the ACDA system.

The role of the communication component is to connect modules of the ACDA system through wired and wireless links. Three types of wireless communication links in the ACDA system include the Air to Ground (A2G) / Ground to Air (G2A), Air to Air (A2A), and

Air to Computing module (A2C) / Computing module to Air (C2A) links. We adopt multiple communication technologies to avoid the mutual interference. The role of the computing component is to collect data from sensors and the other components, process the computing and learning tasks, and then output the motor control signal to the control component. The role of the control component is to drive the motor that rotates the directional antennas for autonomous alignment and form a robust long-distance communication channel.

Figure 3 shows the modules of each component and their connections of the subsystem on one side. The communication component includes a directional antenna, a Wi-Fi router, and a Wi-Fi adapter. The computing component is realized on a microprocessor. The control component contains a rotating motor, a tunable plate, a motor driver, and a compass module. The user interface installed on the laptop is connected to the communication component through a Wi-Fi router. The GPS module on the UAV platform is connected to the microprocessor.

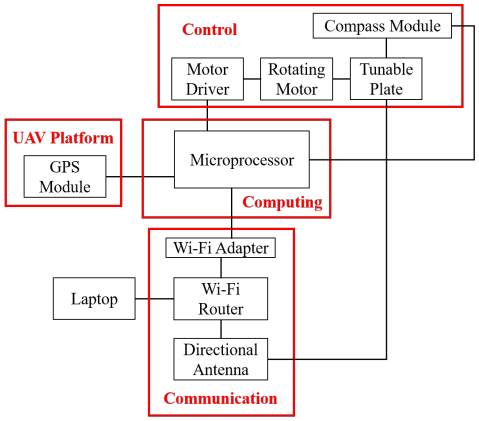


Fig. 3: The components of the subsystems at one side of the ACDA system.

To facilitate practical use of the ACDA system in emergency scenarios, a water-resistant enclosure is designed to protect the core components. Please refer to Figure 4 for photographic illustration of the ACDA prototype.

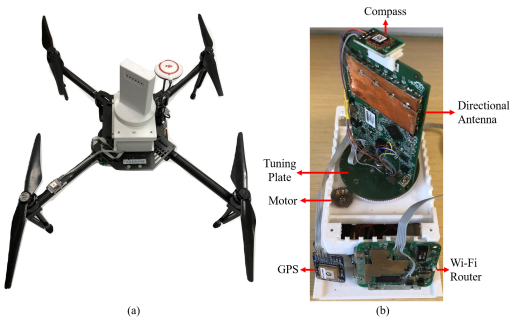


Fig. 4: (a) A picture of the implemented ACDA prototype. (b) Some internal modules of the ACDA system.

2.2 *The UAV Platform*

The UAV platform carries the ACDA system and also performs various flight missions. Considering the performance metrics such as payload, flight time, expandability, stability, and operability, we select DJI Matrice 100, a quad-copter, as our UAV platform (see Figure 4). An alternative candidate of the UAV platform is Tarot 650, which also has good performance as we analyzed in [14]. The features of the DJI Matrice 100 that match our application needs are summarized as follows [23].

(1) *Payload and Flight Time*: Matrice 100 is made of durable and lightweight carbon fiber. The maximum takeoff weight of DJI Matrice 100 is 3.6kg, larger than the total weight of the system (3.13kg), including the platform, the ACDA prototype, and a TB48D battery. The TB48D, a LiPo 6S battery, can support a flight time of about 18 minutes based on our flight tests.

(2) *Expandability*: The expansion bay on the top of Matrice 100 nicely installs the onboard components. In addition, Matrice 100 provides universal power and communication ports, including dual parallel CAN ports and dual UART ports. Matrice 100 is also programmable using the DJI Software Development Kit (SDK), which supports Linux, ROS, Qt and STM32 development environments.

(3) *Stability*: Matrice 100 uses the N1 autopilot, which maintains a stable flight including hovering, under up to 10m/s wind disturbances. Each of the arms incorporates a dampening component that eliminates the vibrations generated from the motors, to improve the stability of the system. In addition, the landing pads are installed at the base of each arm to protect Matrice 100 during landing.

(4) *Operability*: Like other commercial UAVs, the DJI Matrice 100 is ready to fly once open box. The remote controller adopts a simple design. All parameters can be monitored and tuned via a mobile app called “DJI GO”. This app supports a live monitoring of control parameters and real-time videos from any connected smart phones. Some critical information including battery time and GPS signal strength are displayed.

We install a GPS module, Adafruit ultimate GPS breakout version 3 on the DJI Matrice 100 frame to support the ACDA system. It is a high-quality GPS module that can track up to 22 satellites on 66 channels with a high-sensitivity, high-speed receiver and a built-in antenna. In addition, it has the data-logging capability that can record up to 16 hours of data using a FLASH memory.

2.3 *The Communication Component*

The communication component is crucial for the ACDA system, which includes both intra- and inter- device links. The communication links in the ACDA system include the wired links between the directional antenna and the Wi-Fi router, and the wireless links for the A2A, A2G, G2A, A2C, and C2A communications (see Figure 5).

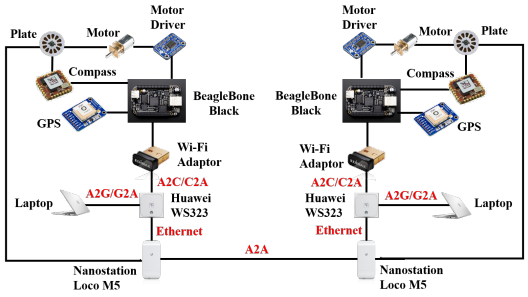


Fig. 5: Illustration of the connections among different components of the ACDA system, with a highlight on the communication links.

(1) *A2A Link*: We select Ubiquiti Nanostation Loco M5 [24] as the directional antenna unit to establish the A2A communication link, based on an evaluation of the performance, scalability, cost, size and weight. Utilizing the 2×2 multiple-input and multiple-output (MIMO) technology, Loco M5 is a small form factor unit operating at the 5 GHz band with default antenna gain of 13 dBi. In addition, it adopts an improved time division multiple access (TDMA) technology, enabling the unit to connect multiple clients with low latency. The data rate is up to 150 Mbps and the maximum transmission distance is up to 10 km. Three modes can be selected according to application scenarios, including access point (AP) mode, client mode and wireless distribution system (WDS) mode. In our design, both the application data, and the command and control signals are transmitted between the local and remote subsystems through this A2A link.

(2) *Wired Link*: The Wi-Fi router, Huawei WS323 [25], is connected to Loco M5 through a Ethernet cable. Huawei WS323 supports both 2.4 GHz and 5 GHz bands and complies with the IEEE 802.11n standard. It adopts a 2×2 MIMO and delivers a wireless transmission rate of up to 300 Mbps while simultaneously providing the multi-device access capabilities.

(3) *A2G/G2A Links*: Data from both the local and remote UAVs are transmitted to the ground in real time using the A2G link. We use a laptop on the ground on each side to receive the application data and diagnosis information of the system, which are displayed on a user-friendly interface (to be described in Section 2.8). The laptop is connected to Huawei WS323 through the wireless local area network (WLAN). The G2A link is a reverse process of A2G. By employing the interface, user control and command signals can be sent to the Huawei WS323 located on the corresponding UAV, and then Huawei WS323 relays the information to their destinations in the network.

(4) *A2C/C2A Links*: Data are transmitted to the microprocessor for processing through the A2C link. Beaglebone Black, the microprocessor, is responsible for all the computing and learning tasks based on data received through a Wi-Fi adapter from other components in the network. The C2A Link is a reverse process of the A2C link. The microprocessor sends processing results to its destinations in the network through the Wi-Fi adapter.

Other than the aforementioned wireless links designed for the ACDA system, the operation of the ACDA system also relies on several other wireless links. Altogether, there are five types of wireless links required for the ACDA system: 1). GPS signal link (1575MHz) which receives the location information from GPS satellites, 2). flight control signal link (2.4GHz) which transmits control commands of the remote controller to the corresponding UAV, 3). A2A link (5GHz) which connects the local and remote subsystems of the ACDA system, 4). A2G/G2A link (2.4GHz/5GHz) connects the laptops to the ACDA network, and 5). A2C/C2A link (2.4GHz) which connects the microprocessors to the ACDA network.

We here analyze the aforementioned wireless links and provide a configuration strategy to effectively utilize different frequency bands to avoid the mutual interference among the communication links. The GPS signal operates on a frequency which doesn't cause the interference with other links. Since BeagleBone Black supports only 2.4GHz, the A2C/C2A links can only operate on the 2.4GHz frequency band, and hence we configure the Huawei WS323 to operate on 2.4GHz. Because the A2C/C2A and the A2G/G2A are supported by the same Wi-Fi router, i.e., Huawei WS323, A2G/G2A link also operates on 2.4GHz. The A2G/G2A, A2C/C2G, and the flight control signal link share the 2.4GHz frequency band, which does not interfere with the A2A link. Figure 6 shows the frequencies and bandwidths of the wireless links in the ACDA system.

To avoid the mutual interference among Huawei WS323, the flight control signal link, and the communication environment, we develop an algorithm to automatically select the Wi-Fi channels for Huawei WS323 and the flight control system of DJI Matrices 100 as shown in Algorithm 1.

The communication links in the ACDA system are configured as follows.

(1). *Ubiquiti Nanostation Loco M5*: The network modes of both the local and remote Nanostation Loco M5 are set up as a bridge, which is a point-to-point link. The wireless mode of the local Nanostation Loco M5 is chosen as a client, and the remote Nanostation Loco M5 is chosen as an access point. The same WLAN service set identifier (SSID) is selected for both the local and the remote sides. The output power is set as 23 dBm. We choose the most secure option WPA2-PSK (AES), which applies the Wi-Fi protected access II algorithm with the advanced encryption standard. Channel width is set to 40MHz. The WLAN IP addresses for both sides are randomly selected in the same subnet. For example, in our system, we set the local Nanostation IP address as 192.168.33.100, and the remote IP address as 192.168.33.110.

(2). *Huawei WS323*: Huawei WS323 is configured to operate on the 2.4GHz frequency band. The channel width is set to 40MHz and the WLAN communication standard of 802.11n is selected. We select different SSIDs for the local Huawei router and the remote Huawei router, which both differ from the SSID of Ubiquiti Nanostation Loco M5. The IP addresses of them should also be in the same subnet as the Nanostations. For example, we set the IP address of local Huawei router as 192.168.33.111, and the remote Huawei router as 192.168.33.121.

(3). *Wi-Fi adapter*: The IP addresses of the local and remote Wi-Fi adapters are set under the same subnet as Nanostations and Huawei routers, i.e., 192.168.33.130 for the local Wi-Fi adapter and 192.168.33.131 for the remote Wi-Fi adapter.

(4). *Laptop*: Both the local and remote laptops are set under the same subnet as other communication modules in the ACDA system, i.e., 192.168.33.21 for the local laptop and 192.168.33.122 for the remote laptop.

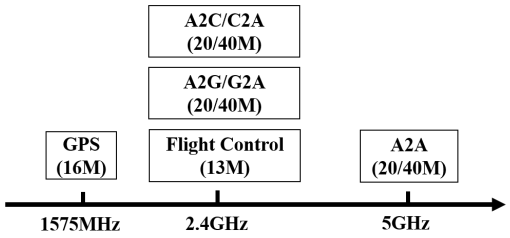


Fig. 6: The frequencies and bandwidths of the wireless links of the ACDA system.

Algorithm 1 Wi-Fi Channel Selection Algorithm

Requirement: An ACDA system placed in a given environment.
Output: Wi-Fi channels selected for both the local and remote Huawei WS323 and the flight control system of DJI Matrices 100.

Procedure:
1. Scan wireless signals and record their frequencies, channels, and RSSIs for the local subsystem.
2. Find the least crowded channel with low RSSIs and select it as the channel for local Huawei WS323.
3. Find another uncrowded channel with low RSSIs which does not overlap with local Huawei's channel, and select it as the channel for the local flight control of DJI Matrices 100.
4. Repeat the procedure 1 to 3 for the remote subsystem.

2.4 The Computing Component

The computing functionality is implemented on a microprocessor. We select BeagleBone Black, a low-cost, community-supported development platform featuring a 32-bit RISC microprocessor as the computing component [21]. It has a powerful CPU (ARM Cortex-A8) with up to 1 GHz clock time, compared to the 16 MHz clock time for the Arduino used in the previous version of ACDA. It has 512MB DDR3 RAM, 4GB 8-bit eMMC onboard flash storage,

3D graphics accelerator, NEON floating-point accelerator, and two programmable real-time unit (PRU) 32-bit micro-controllers. The BeagleBone Black supports various ports, for example, USB, Ethernet, HDMI, and 2×46 pin headers, making it flexible for our ACDA design. In addition, it's compatible with the Linux distributions Debian and Ubuntu, which serve as the platform for ROS, an open-source and meta-operating system. It also supports multiprocessing, which is very useful to transmit multiple sensor data streams with devices in the network. BeagleBone Black does not have a wireless card assembled, and hence we attach a portable Wi-Fi adapter to enable its communication with other components of the ACDA system.

The data processing in BeagleBone Black uses three types of data from the other components, including the GPS on the UAV platform, compass data in the control component, and RSSI retrieved from Nanostation Loco M5 through a Wi-Fi adapter. The output is the motor control signal to the control component.

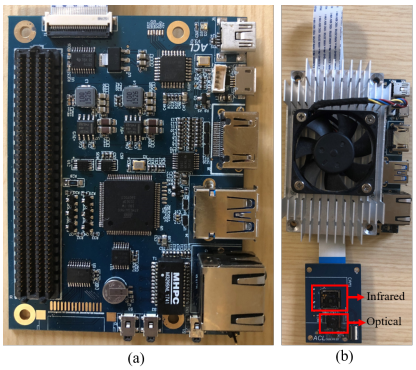
2.5 The Antenna Heading Control Component

The antenna heading control component includes a motor driver, a rotating motor, a tunable plate, and a compass module (see Figure 4 (b)). The directional antenna and the compass module are placed on a tunable plate, which is controlled by a motor. The motor driver receives motor control signals from the microprocessor and outputs the pulse width modulation (PWM) signals to rotate for a specified angle.

We select Adafruit TB6612 as the motor driver. The rotating motor is a micro-gearmotor with a maximum speed of 130 revolutions per minute (RPM) and a gear ratio of 210:1. The tunable plate is a plastic robot gear. We use MTI-3-8A7G6T Xsens as our compass module. It has a full 3D magnetometer-enhanced attitude and heading reference system (AHRS) with an in-run compass calibration (ICC) setting, which can compensate for magnetic distortions. In addition, it provides an active heading stabilization (AHS) setting, which can significantly reduce the heading drift under magnetic disturbances. The antenna heading control algorithm to align the directional antennas for a robust A2A channel is designed in Section 3.

2.6 Cameras in the ACDA system

The emergency response application requires both optical and infrared monitoring videos of the remote disaster zone to be transmitted to the ground station in real-time.



(a) (b)

Fig. 7: (a) Carrier board design. (b) Jetson TX2 carrier board with two cameras, i.e., the infrared camera and the optical camera.

On the remote side, the UAV carries a NVIDIA Jetson TX2 module that processes data from the two video sensors OV5693 and FLIR Lepton 2 (see Figure 7). As Jetson TX2 has a large development board, we developed a carried board of TX2 with weight 53g and size 88mm×65mm, suitable for UAV applications (see Figure 7a). OV5693 is a 1/4-inch, 5-megapixel image sensor, which delivers full 1080p high-definition video streams at 30 frames per second (fps). FLIR Lepton 2, a complete long-wave infrared (LWIR) camera, is connected to Jetson TX2 to capture infrared radiation and output a uniform thermal image. Jetson TX2 supports Ubuntu, and hence can run ROS. Jetson TX2 is connected to the remote Wi-Fi router, i.e. Huawei WS323 through WLAN with an IP address 192.168.33.133. The local ground station receives the two video streams from the remote UAV in real time through the A2A link.

2.7 ROS and Information Flow

We adopt ROS to support data transmissions in the ACDA system. Here we describe ROS, and the ACDA information flow built on ROS.

ROS is a widely used middleware for developing robot applications [26]. It's a collection of tools, libraries, and conventions that enable users to build various applications by providing services such as low-level device control, message-passing between processes, and package management. It works well on traditional operating systems, such as Ubuntu in our case, and can support multi-device communication, distributed computation, and rapid testing.

ROS is composed of packages. A ROS package is a collection of files that serves for a specific purpose, generally including source files, executable files and supporting files (libraries, configuration files, dataset, etc.). ROS nodes are a set of basic independent executable units that perform various tasks. In order to share information, all nodes must communicate with each other. The primary mechanism that ROS nodes adopt for communication is fto send and receive messages. A message is a simple data structure consisting of typed fields. Messages are organized into named topics. A node shares information by publishing messages on appropriate topics, and a node receives information by subscribing to the corresponding topics. The nodes do not need to know whom they are communicating with, but only publish or subscribe messages of the topics of interest [26]. ROS allows multiple publishers and subscribers for a topic. To facilitate the communication among nodes, a ROS master provides naming and registration services to the ROS nodes. The ROS master tracks publishers and subscribers of topics and enables individual ROS nodes to locate each others.

The use of ROS nodes provides several benefits to the design and implementation of the ACDA system. First, the publish/subscribe mechanism simplifies the data transmission among multiple components in a network. Second, the system is fault tolerant as crashes are isolated to individual nodes. Third, code complexity is reduced by encapsulating small tasks into nodes and assembling them in a structured way. Forth, implementation details are hidden as the nodes expose a minimal application programming interface (API) to the rest of the system and hence can be easily modified or replaced.

Our ACDA system has the following ROS architecture. The ROS master runs on the local BeagleBone Black. It has four ROS clients, including the remote BeagleBone Black, the Jetson TX2 to support video processing, the local laptop, and the remote laptop. The ROS topics published and subscribed by the ROS master and clients are shown in Figure 8.

The messages in the system include GPS, RSSI, compass, and camera information. Each message here corresponds to a specific topic. The local BeagleBone Black publishes the local compass message, GPS message, RSSI message and heading angle message, and subscribes the remote GPS message and remote RSSI message to conduct the RL-based on-line directional antenna control algorithm at the local side. The remote BeagleBone Black functions in a similar way. The remote Jetson TX2 publishes two camera messages including the infrared video message and optical video message, so that the videos on the remote side can be tracked at the ground stations on the local side. The local laptop and the remote laptop subscribe all the topics in the ACDA system and display the information on a user-friendly interface to monitor the system performance and conduct various operations.

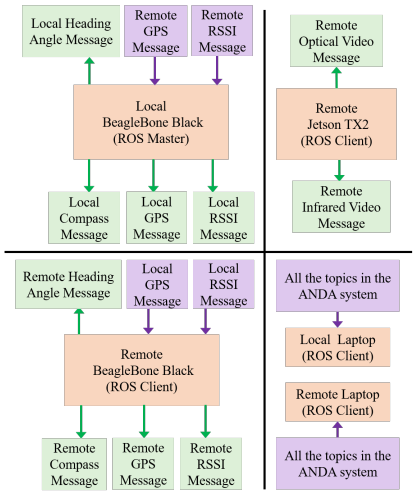


Fig. 8: ROS architecture and topics in ACDA system. The orange blocks denote the the ROS hosts, including the ROS master and ROS clients. The green blocks denote published messages and the purple blocks denote subscribed messages.

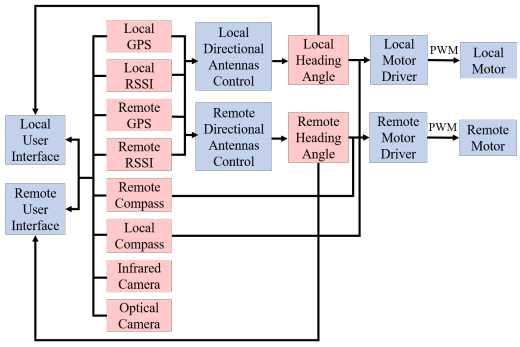


Fig. 9: Information flow of the ACDA System. The red blocks denote messages in the system. The blue blocks denote components to which these information flow.

Figure 9 shows the ROS information flow. The local directional antenna control algorithm running on a microprocessor calculates the local antenna heading angle by using both the local and remote GPS and RSSI signals, and sends the PWM signals to the local motor driver. The information flow on the remote side is similar. The local and remote user interfaces receive and display all the messages in the ACDA system.

2.8 *Interface Design*

A user-friendly interface (see Figure 10) is designed to automate the configuration, operation, and diagnosis procedures. The interface displays the locations of UAVs, communication quality, antenna alignment performance, and real-time videos. In addition, control operations are included to automate the intricate configuration processes, lowering the burden for users such as emergency staff.

The graphical user interface (GUI) is developed using Qt, a cross-platform application framework and widget toolkit [22]. Qt supports all major desktop platforms and most mobile or embedded platforms. Its advantages include fast and simple programming, consistent and comprehensive APIs and libraries, and compatibility with various compilers.

Fig. 10: A user-friendly interface design.

The interface of the ACDA system is composed of four parts, including the message input and display, map, infrared and optical video display, and control panel. The message input and display support two functions. First, it displays information of the local and remote compasses, GPS, RSSIs and antenna headings retrieved from the onboard ACDA system. The local antenna heading angle is marked on a compass dial in red, and the remote antenna heading is marked on the compass dial in blue. The GPS, RSSI and desired antenna heading angles are displayed in text boxes. In addition, two circle indicators on the right side of the compass dials show the connection qualities of the ACDA system on the local side and the remote side respectively, where green denotes connection established and red denotes connection lost. Second, the text boxes of GPS and antenna heading angles can also serve as input boxes. If GPS or desired heading angle is known in advance, users can input it directly to the corresponding text box and pressing "Enter".

The map displays the geographic information of the local and remote UAVs using OpenStreetMap (OSM), an open source mapping toolkit. The red UAV icon represents the location of the local UAV, and the blue UAV icon represents the location of the remote UAV. The distance between them is displayed on the upper left corner of the map. The video display shows the real-time infrared and optical videos transmitted from the remote side. Some function keys can be used to adjust the images, such as refresh, resize and rotate. The control panel consists of six control operations, each having several widgets that facilitate user operations illustrated below:

(1) *Distance Mode Selection.* There are two modes: line-of-sight (LOS) and NLOS. In the LOS mode, the local and remote ACDA subsystems are within a short range (100-200m) and there is no obstruction between them, and hence the communication of the ACDA system can be established without alignment. In the NLOS mode, either the distance between the UAVs is beyond the short range or the signals of the two systems are blocked by obstructions in between. When LOS is chosen, the antennas alignment can be completed on the ground for optimal communication performance. When NLOS is chosen, the antenna alignment starts after UAVs are launched to the air, because attenuation, reflection, diffraction, and penetrations do not permit the alignment on the ground.

(2) *Channel Selection.* Once the "Optimize" button is clicked, the Wi-Fi channel selection algorithm described in Algorithm 1 is implemented and the optimized channels which avoid mutual interference selected on the local and remote sides are displayed in the text box to the right.

(3) *IMU Calibration.* The IMU calibration is automated. Its purpose is to eliminate magnetic interference of the environment. Once the "Run" button is clicked, the IMU calibration begins, "Run" becomes "Stop", and the text box to the right shows "Working".

One can stop the IMU calibration at any time by clicking the button again. When the IMU calibration is complete, the text box to the right shows "Done".

(4) *Initial Heading Selection.* This setting is used in a GPS denied environment such as indoors to align directional antennas by exhausting various pairs of heading angles at the local and remote sides to find the maximum RSSI. The button has two functions, "Scan" and "Stop". The text box to the right shows three states: "Default", "Working", and "Done". "Default" means that the GPS is available in current scenario, so that the initial heading selection step can be skipped. "Working" and "Done" are shown when the initial heading scan is triggered and completed, respectively.

(5) *ROS Node Operations*. Similar to the button for IMU calibration, the button of ROS nodes has two functions: "Run" and "Stop", with corresponding states, "Working" or "Done", displayed in the text box to the right. Once the ROS nodes are activated, topics are created and messages are exchanged among functional modules of the local and remote ACDA subsystems.

(6) *Cameras in Use.* Four check boxes are included for the infrared and optical cameras. Once a camera is checked, the real-time video captured by this camera will be displayed on the right. When "Save" is checked, the video will also be stored in the local laptop.

Figure 11 shows the flowchart of user operations.

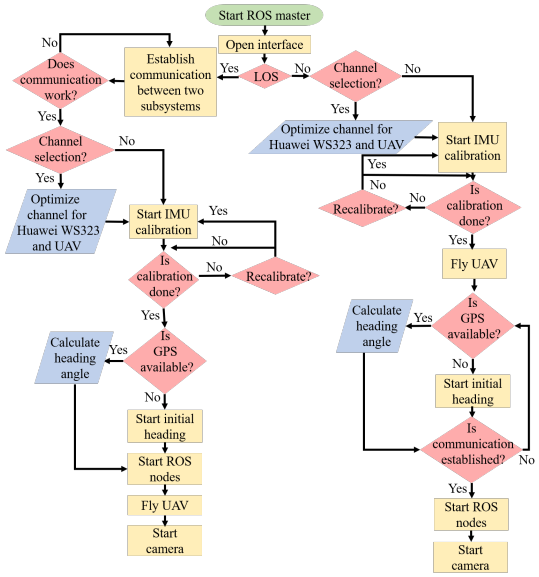


Fig. 11: Flow chart of operations for the ACDA, coded in the user interface.

2.9 *Water-Resistant Enclosure Design*

The water-resistant enclosure design of the ACDA system is included for the emergency response operations (see Figure 12).

The water-resistant enclosure consists of two parts: the cover for the antenna which can rotate a full 360 degrees, and the cover for the other components, including battery, communication, computing, and control components. It supports both static seal such as static radial seal, and dynamic seal such as rotary seal. The enclosure is printed using a 3D printer with Acrylonitrile Butadiene Styrene (ABS) plastic.

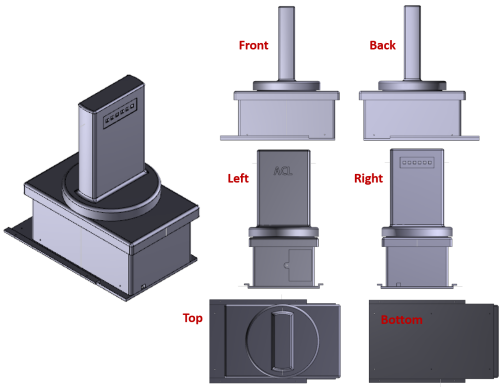


Fig. 12: Water-resistant enclosure design.

3 The ACDA System Model and Controller Design

In this section, we first describe the ACDA system model including the UAV and directional antenna dynamics. We then describe the measurement models for GPS and RSSI. The antenna control problem is then formulated and a RL-based solution in an unknown communication environment is developed in the end.

3.1 *System Models*

3.1.1 UAV Dynamics: Two UAVs operate at approximately the same height. Denote the positions of the UAV i along x and y axes at time instant k as $x_i[k]$ and $y_i[k]$ respectively. The dynamic of UAV i (denote as $f_i(.)$) is described as

$$\begin{aligned} x_i[k+1] &= x_i[k] + v_i[k]\cos(\phi_i[k])\delta, \\ y_i[k+1] &= y_i[k] + v_i[k]\sin(\phi_i[k])\delta, \\ \phi_i[k+1] &= \phi_i[k] + \omega_i[k]\delta, \end{aligned} \tag{1}$$

where δ is the sampling period, $\phi_i[k]$ and $\omega_i[k]$ are the heading angle and angular velocity of UAV i at time instant k respectively.

3.1.2 Directional Antenna Dynamics: The directional antenna installed on each UAV adjusts its heading angle automatically to establish the robust A2A communication channel. The heading angle dynamics of its directional antenna is described as

$$\theta_i[k+1] = \theta_i[k] + (\omega_i^*[k] + \omega_i[k])\delta, \tag{2}$$

where θ_i is the heading angle of antennas i, and ω_i^* is the angular velocity of antennas i due to its heading control. Note that both the control of antennas i (ω_i^*) and the movement of UAV i (ω_i) contribute to the change of θ_i .

3.2 *Measurement Models*

Two measurement models are included, one for GPS, and the other for RSSI. The communication performance indicator RSSI is adopted to assist with the distributed antenna controller design.

3.2.1 GPS measurement: The measurement of the GPS signal $Z_{G,i}(k)$, is described as

$$Z_{G,i}[k] = H_{G,i}(k)X_i[k] + \varpi_{G,i}[k], \tag{3}$$

where H_G is the measurement matrix, $H_{G,i} = [1,0,0,0;0,1,0,0]$. $X_i[k] = [x_i[k], y_i[k], \phi_i[k], \theta_i[k]]^T$ is the system state of UAV i,

and $\varpi_{G,i}$ is the white Gaussian noise with zero mean and covariance $R_{G,i}$. GPS signals can be transmitted through the A2A communication channel to assist with the control of the directional antennas.

3.2.2 RSSI measurement: RSSI measures the performance of the communication channel [18, 19]. In the ACDA system that is equipped with two directional antennas, RSSI is affected by 1) the relative positions of the two UAVs that carry these directional antennas, 2) the headings of these antennas, and 3) the field radiation patterns of these antennas in a specific communication environment. The RSSI signal $Z_R[k]$ can be derived from the Friis free space equation as [27]:

$$\begin{aligned} Z_R[k] =& P_{t|dBm}[k] + G_{l|dBi} + 20\log_{10}(\lambda) \\ & - 20\log_{10}(4\pi) - 20log_{10}(d[k]) + \varpi_R[k], \end{aligned} \tag{4}$$

where $P_{t|dBm}[k]$ is the transmitted signal power, λ is the wavelength, $d[k]$ is the distance between the two UAVs at time k, and $d[k] = \sqrt{(x_1[k]-x_2[k])^2 + (y_1[k]-y_2[k])^2}$. $\varpi_R[k]$ is the white Gaussian noise with zero mean, and $G_{l|dBi}[k]$ is the sum of the transmitting and receiving antenna gains [28]. For the Ubiquiti NanoStation loco M5 directional antennas [24] that we use in the ACDA system, it is modeled based on the filed pattern of the end-fire array antennas [29]

$$\begin{aligned} G_{l|dBi}[k] =& (G^{max}_{t|dBi} - G^{min}_{t|dBi}) \\ & \times \sin\frac{\pi}{2n}\left|\frac{\sin(\frac{n}{2}(k_a d_a(\cos(\gamma_t[k]-\theta_t[k]))-1)-\frac{\pi}{n})}{\sin(\frac{1}{2}(k_a d_a(\cos(\gamma_t[k]-\theta_t[k]))-1)-\frac{\pi}{n})}\right| \\ & + (G^{max}_{r|dBi} - G^{min}_{r|dBi}) \\ & \times \sin\frac{\pi}{2n}\left|\frac{\sin(\frac{n}{2}(k_a d_a(\cos(\gamma_r[k]-\theta_r[k]))-1)-\frac{\pi}{n})}{\sin(\frac{1}{2}(k_a d_a(\cos(\gamma_r[k]-\theta_r[k]))-1)-\frac{\pi}{n})}\right| \\ & + G^{min}_{t|dBi} + G^{min}_{r|dBi}, \end{aligned} \tag{5}$$

where $G^{max}_{t|dBi}$, $G^{min}_{t|dBi}$, and $G^{max}_{r|dBi}$, $G^{min}_{r|dBi}$ are the maximum and minimum gains of transmitting and receiving antennas. k_a is the wave number, and $k_a = \frac{2\pi}{\lambda}$. n and d_a are parameters decided by the design of the antenna. $\theta_t[k]$ and $\theta_r[k]$ are the heading angles of the transmitting and receiving antennas at time k. $\gamma_t[k]$ and $\gamma_r[k]$ are the heading angles of the transmitting and receiving antennas corresponding to the maximal $G_{l|dBi}[k]$ at time k.

The parameters $G^{max}_{t|dBi}$, $G^{min}_{t|dBi}$,$G^{max}_{r|dBi}$, and $G^{min}_{r|dBi}$ can be obtained from the antenna's data sheet in an ideal environment. In our ACDA system, $G^{max}_{t|dBi} = G^{max}_{r|dBi}$ and $G^{min}_{t|dBi} = G^{min}_{r|dBi}$ hold because the two directional antennas are of the same type. In an imperfect environment where disturbances and interference exist, these parameters in $G_{l|dBi}[k]$ can be environment-specific.

In addition, $\gamma_t[k]$ and $\gamma_r[k]$ can be obtained from the alignment of the two directional antennas [20]. In an imperfect environment, such as blockages, the desired heading angles can be captured by

$$\gamma_r[k] = \arctan\frac{y_t[k]-y_r[k]}{x_t[k]-x_r[k]} + \theta_{env}, \tag{6}$$

where $(x_t[k], y_t[k])$ and $(x_r[k], y_r[k])$ are the positions of UAVs that carry the transmitting and receiving antennas respectively, and θ_{env} is an environment-specific shift angle. $\theta_{env} = 0$ in a perfect environment.

3.3 Problem Formulation

Our goal is to find the directional antennas' optimal heading angle velocities to maximize the RSSI performance over a look-ahead window, and thus, to maximize the performance of the UAV-to-UAV communication channel. The RSSI model (as described in Equations (4)-(6)) contains unknown environment-specific parameters, i.e., $G^{max}_{t|dBi}$, $G^{min}_{t|dBi}$, and θ_{env}.

Here we formulate the problem as an optimal control problem. Mathematically, considering the system dynamics described in Equations (1) and (2), an optimal control policy $U[k]$ is sought to maximize the following value function

$$V(X[k]) = \sum_{l=k}^{k+N} \alpha^{l-k} Z_R[l](X[l], U[k]), \tag{7}$$

where $X[k]$ is the global state, $X[k] = [X_1^T[k], X_2^T[k]]^T$, $X_i[k] = [x_i[k], y_i[k], \phi_i[k], \theta_i[k]]^T$. $U[k]$ is the control input, $U[k] = [U_1[k], U_2[k]]^T$, $U_i[k] = [\omega_i^*[k]]$. $\alpha \in (0, 1]$ is a discounter factor, and $Z_R[l]$ is the RSSI signal at time l. Note that the control is decentralized. Each directional antenna finds its own optimal control policy with the assumption that the other antenna adopts its optimal control policy.

Next we develop the control solution for one of the UAVs (denoted as the local UAV, or UAV 1). The control solution for the other UAV (the remote UAV, or UAV 2) is designed in the same manner.

3.4 Reinforcement Learning based Optimal Control for ACDA

To solve the optimal control problem formulated above, an on-line adaptive optimal controller is developed based on reinforcement learning.

Note that the value function (described as Equation (7)) can be rewritten as

$$\begin{aligned} V(X[k]) =& Z_R[k](X[k], U[k]) \\ & + \sum_{l=k+1}^{k+N} \alpha^{l-k} Z_R[l](X[l], U[k]) \end{aligned} \tag{8}$$

To solve the above Equation online, here we utilize the RL method, in particular, the policy iteration (PI) algorithm [30, 31]. The PI algorithm contains two steps: policy evaluation and policy improvement. The policy evaluation step is designed to solve the value function $V(X[k])$ using Equation (8), given the current control policy. The policy improvement step is to maximize the value function by finding the optimal control policy. The two steps are conducted iteratively until convergence.

Policy Evaluation

$$\begin{aligned} V_{j+1}(X[k]) =& Z_R[k](X[k], U[k]) \\ & + \sum_{l=k+1}^{k+N} \alpha^{l-k} Z_{j,R}[l](X[l], U[k]) \end{aligned} \tag{9}$$

Policy Improvement

$$\begin{aligned} U_{j+1}(X[k]) =& \arg\max_{U_j[k]} Z_R[k](X[k], U[k]) \\ & + \sum_{l=k+1}^{k+N} \alpha^{l-k} Z_{j+1,R}[l](X[l], U[k]) \end{aligned} \tag{10}$$

where j is the iteration step index, and $Z_{j,R}[l](X[l], U[k])$ is the RSSI model with parameters learned in the jth iteration step.

Note that three unknown parameters for the environment-specific RSSI model ($G^{max}_{t|dBi}$, $G^{min}_{t|dBi}$ and θ_{env}) are involved in Equation

(9), and need to be learned. In particular, for each iteration j, three time steps (k, $k+1$ and $k+2$) are needed to come up with three equations to iteratively solve for the three parameters. To solve the nonlinear equations, we utilize the Newton's method [32]. The idea of the Newton's method is described as follows. An initial guess which is reasonably close to the true root is first given, and the function is then approximated at the given initial guess by its tangent line. After computing the x-intercept of this tangent line, which is typically a better approximation to the function's root than the original guess, the function is approximated again at the derived x-intercept until the accuracy meets the requirement.

4 Experimental Results

4.1 Simulation Studies

In this section, we conduct simulation studies to illustrate and validate the antenna controller design. Two UAVs are simulated to move in a 2-D airspace, each equipped with a directional antenna. The total simulation time is $T = 45s$, with the sampling period $\delta = 1s$. The parameter N in Equation (7) is selected as 1 here (i.e., with the goal of maximizing the current RSSI). The transmitting power is $P_{t|dBm} = 23dBm$, and the electromagnetic wavelength is $\lambda = 0.052m$. For the parameters in Equation (5), we select the design parameters $n = 8$ and $k_a = \frac{\lambda}{4}$. Figures 13(a) and 13(b) show the trajectories of local UAV and remote UAV respectively. The blue solid curve and the red dotted curve are the real trajectory and the GPS measurements respectively. Gaussian noises are added to the GPS signals.

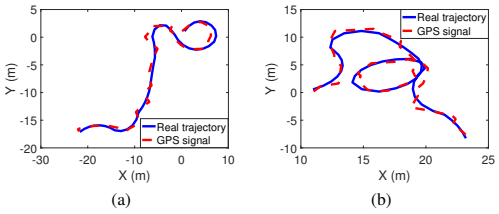


Fig. 13: (a) Trajectory of UAV 1, and (b) trajectory of UAV 2.

With these randomly generated UAV trajectories, we simulate the RL-based control algorithm. To simulate the long-distance communication scenario, the minimum received signal strength is assumed to be 0, and in this case, the minimum directional antennas' gain ($G^{min}_{t|dBi}$) can be calculated accordingly. Figures 14(a) and 14(b) show the learned environment-specific antennas' maximum gain ($G^{max}_{t|dBi}$) and the shift angle caused by the environment (θ_{env}) respectively. Gaussian noises are added to the RSSI measurements. As shown in the figures, the learned parameters are very close to their true values, which indicates the effectiveness of the learning algorithm. Figures 15(a) and 15(b) show the derived optimal heading angles of the local directional antenna and the angle errors between the derived optimal heading angle and the true optimal heading angles. The derived optimal heading angles are very close to the true optimal angles, indicating the good performance of the RL-based control algorithm.

4.2 Field Tests

We conducted two field tests to verify the proposed ACDA solution. The first is to verify the RSSI model, and the second is to test the performance of the antenna control algorithm through a comparative study with the sole GPS-based control algorithm described in [14].

4.2.1 Testing of the RSSI Model: We first test the RSSI model for a pair of directional antennas, as described in Equations (4) and

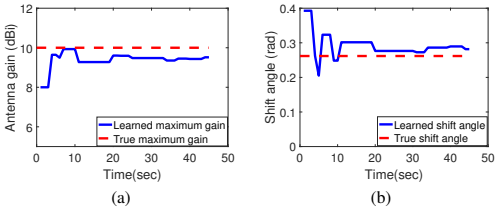


Fig. 14: Learned environment-related (a) maximum directional antenna gain, and (b) shift angle.

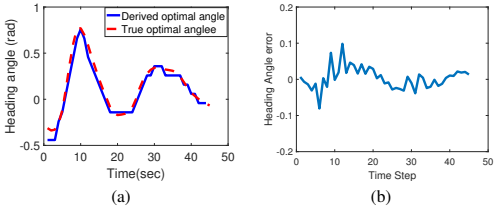


Fig. 15: (a) Derived optimal heading angles. (b) Heading angle errors between the derived angles and the true optimal angles.

(5). The distance between the two directional antennas is $43m$, the transmitting power is $23dBm$, and the electromagnetic wave frequency is $5.8GHz$. With the same environment and test settings, we conduct two tests to check the properties of the two directional antennas. In the first test, the remote antenna points towards the local antenna and keeps a fixed heading angle throughout the test. The local antenna initially points towards the remote antenna and then rotates 15° per $90s$ until 360°. The RSSI is collected by the local antenna and is averaged during each $90s$. In the second test, the heading angle of the local antenna is fixed, and points towards the remote antenna. The remote antenna initially points towards the local antenna and rotate 15 degrees per 90 seconds until 360°. The RSSI collected by the remote antenna is also averaged in the second test. Parameters in the RSSI model are estimated from the measured data: $n = 8$, $d_a = \frac{\lambda}{4}$, $G^{max}_1 = 11.5dBi$, $G^{min}_1 = -8.5dBi$ for the local antenna, and $G^{max}_2 = 9dBi$, $G^{min}_2 = -8dBi$ for the remote antenna. Figures 16(a) and 16(b) show the test results. The tested RSSI mappings match reasonably with the proposed model.

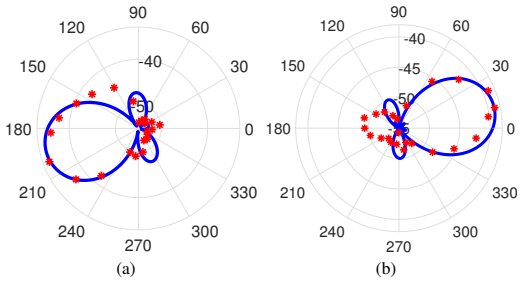


Fig. 16: RSSI mapping in (a) the first test, and (b) the second test. The red dots are the measured RSSI signal, and the blue curves pictures the relation between the RSSI signal and the rotation angle according to Equations (4) and (5).

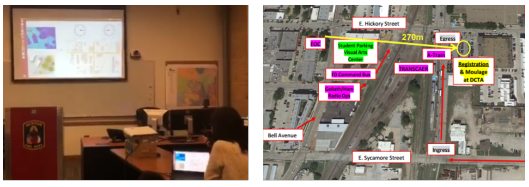
4.2.2 Testing of the control algorithm : We also test the proposed RL-based control algorithm. In this preliminary testing, the local directional antenna is at a fixed location, and the remote antenna changes its location every $90s$. Each antenna finds its own optimal heading angle at each location according to the distributed RL-based control algorithm described in Section 3 to optimize the RSSI performance. The RSSI is measured at both antennas with the derived heading angles. To provide a comparison, we also test the GPS alignment-based control algorithm proposed in [14], in which each antenna points toward the GPS location of the other controller, and RSSI is not used as a measurement signal. The testing results are summarized in Table 1. At each location, two control algorithms are tested. "RL" in Table 1 represents the RL-based control algorithm, and "GPS" means the GPS alignment-based control algorithm.

Table 1 Comparative Field Testing Results

		Local antenna		Remote antenna	
Position	Control	RSSI	Heading	RSSI	Heading
1	RL	-37dBm	194.1°	-41dBm	16.4°
	GPS	-45dBm	170.6°	-45dBm	35.5°
2	RL	-37dBm	197.3°	-39dBm	15.1°
	GPS	-39dBm	176.2°	-42dBm	357.8°
3	RL	-39dBm	191.1°	-44dBm	13.2°
	GPS	-41dBm	182.6°	-44dBm	6.2°
4	RL	-35dBm	196°	-39dBm	15.2°
	GPS	-38dBm	195.8°	-39dBm	16.2°
5	RL	-35dBm	194.9°	-39dBm	15.4°
	GPS	-37dBm	186.1°	-39dBm	7.5°

The proposed RL-based algorithm demonstrates a better RSSI, as compared to the GPS alignment-based algorithm, in all five locations. The testing results verifies that the proposed RL-based control algorithm outperforms the GPS alignment-based algorithm.

4.2.3 Emergency Drill: We tested the performance of our ACDA system through participating in a full-scale emergency drill of the City of Denton, TX. On a raining day in May 2018, a full-scale disaster drill on a train-involved accident was conducted with the lead of the Denton Fire Department. An incident occurred on the Union Pacific rail line and passengers on the Denton county transportation authority (DCTA) A-Train piled out of the train with a great panic, and caused injuries and fatalities. Our ACDA system successfully responded to the emergency and sent back real-time monitoring videos of the emergency zone to the emergency management center for condition assessment (see Figure 17).



(a) (b)

Fig. 17: Emergency drill with Denton Fire Department. (a) The room of the emergency management center and the screen where the remote infrared and optical monitoring videos are transmitted and displayed. (b) The map of this disaster drill. The distance between the local and the remote UAVs is about 270m, beyond the nominal Wi-Fi range.

5 Conclusions and Future Works

In this paper, a new long-range broadband ACDA system which seamlessly integrates the communication, control and computing components was designed, implemented and tested. The communication component features a directional-antenna equipped UAV-to-UAV channel that shares the transmissions of the application data, control and commands, and the autonomous directional antenna control that focuses the energy and significantly extends the communication distance with reduced interference. The computing and control components feature the use of RSSI, a communication performance indicator to assist with the antenna control to achieve optimal communication performance, and a RL-based control algorithm that learns the communication model in an unknown environment. We adopted ROS and BBB for the implementation of ACDA, and considered practical issues such as channel selection, interface design, application-oriented sensors and waster-resistant design for the practical use of the ACDA system in emergency response operations. The simulation studies, field tests, and disaster drills verified the performance of the system. The ACDA system can be broadly used to provide on-demand broad-band and long-distance UAV-to-UAV communication and service to the ground in places where communication infrastructures do not exit. In the future work, we will extend the solution to multi-UAV communication, by using phase-array-based electronically controlled directional antennas that have multiple controllable narrow beams.

6 Acknowledgment

We acknowledge National Science Foundation under grant numbers 1730325/1730570/1730589/1730675, 1714519, and 1522458 for the support of this work. We would also like to acknowledge M. A. Penaluna at Denton Fire Department, C. Kessler and R. Davis at Austin Fire Department, D. C. Washington at Denton Municipal Electric, M. McFadden at Emergency Preparedness Department, North Central Texas Council of Governments, R. Smith and S. Keller at Tarrant County Fire Service Training Center, and other emergency personnel for their invaluable contributions to this project.

7 References

1 Wan, Y., Fu, S., Zander, J., Mosterman, P.J.: 'Transforming on-demand communications with drones: The needs, analyses, and solutions', *Homeland Security Today Magazine*, 2015, pp. 32–35
2 Fu, S., Wan, Y.: 'Communicating in remote areas or disaster situations using unmanned aerial vehicles', *HDIAC Journal*, 2016, **2**, pp. 4–8
3 Motlagh, N.H., Bagaa, M., Taleb, T.: 'Uav-based iot platform: A crowd surveillance use case', *IEEE Communications Magazine*, 2017, **55**, pp. 128–134
4 Acevedo, J.J., Arrue, B.C., Maza, I., Ollero, A.: 'Distributed approach for coverage and patrolling missions with a team of heterogeneous aerial robots under communication constraints', *International Journal of Advanced Robotic Systems*, 2013, **10**, pp. 28
5 Pajares, G.: 'Overview and current status of remote sensing applications based on unmanned aerial vehicles (uavs)', *Photogrammetric Engineering & Remote Sensing*, 2015, **81**, pp. 281–330
6 Dixon, C., Frew, E.W.: 'Optimizing cascaded chains of unmanned aircraft acting as communication relays', *IEEE Journal on Selected Areas in Communications*, 2012, **30**, pp. 883–898
7 Dong, X., Hua, Y., Zhou, Y., Ren, Z., Zhong, Y.: 'Theory and experiment on formation-containment control of multiple multirotor unmanned aerial vehicle systems', *IEEE Transactions on Automation Science and Engineering*, 2018, **99**, pp. 1–12
8 Chandrasekharan, S., Gomez, K., Al.Hourani, A., Kandeepan, S., Rasheed, T., Goratti, L., et al.: 'Designing and implementing future aerial communication networks', *IEEE Communications Magazine*, 2016, **54**, pp. 26–34
9 Takahashi, Y., Kawamoto, Y., Nishiyama, H., Kato, N., Ono, F., Miura, R.: 'A novel radio resource optimization method for relay-based unmanned aerial vehicles', *IEEE Transactions on Wireless Communications*, 2018, **17**, (11), pp. 7352–7363
10 Rajashekar, R., Di.Renzo, M., Hari, K., Hanzo, L.: 'A beamforming-aided full-diversity scheme for low-altitude air-to-ground communication systems operating with limited feedback', *IEEE Transactions on Communications*, 2018, **66**, (12), pp. 6602–6613
11 Boehm, F., Schulte, A.: 'Air to ground sensor data distribution using ieee802. 11n wi-fi network', *Proceedings of IEEE Digital Avionics Systems Conference (DASC)*, 2013, pp. 4B2–1–4B2–10
12 Gu, Y., Zhou, M., Fu, S., Wan, Y.: 'Airborne wifi networks through directional antennae: An experimental study', *Proceedings of Wireless Communications and Networking Conference (WCNC)*, 2015, pp. 1314–1319

13 Xie, J., AI.Emrani, F., Gu, Y., Wan, Y., Fu, S.: 'Uav-carried long-distance wi-fi communication infrastructure', *Proceedings of AIAA Infotech@ Aerospace*, 2016, pp. 0747–0760
14 Chen, J., Xie, J., Gu, Y., Li, S., Fu, S., Wan, Y., et al.: 'Long-range and broadband aerial communication using directional antennas (acda): design and implementation', *IEEE Transactions on Vehicular Technology*, 2017, **66**, pp. 10793–10805
15 Dixon, C., Frew, E.W.: 'Maintaining optimal communication chains in robotic sensor networks using mobility control', *Mobile Networks and Applications*, 2009, **14**, pp. 281–291
16 Wang, Z., Giannakis, G.B.: 'A simple and general parameterization quantifying performance in fading channels', *IEEE Transactions on Communications*, 2003, **51**, pp. 1389–1398
17 Wong, K.K., Murch, R.D., Letaief, K.B.: 'Performance enhancement of multiuser mimo wireless communication systems', *IEEE transactions on communications*, 2002, **50**, pp. 1960–1970
18 Min, B.C., Lewis, J., Matson, E.T., Smith, A.H.: 'Heuristic optimization techniques for self-orientation of directional antennas in long-distance point-to-point broadband networks', *Ad Hoc Networks*, 2013, **11**, pp. 2252–2263
19 Min, B.C., Lewis, J., Schrader, D.K., Matson, E.T., Smith, A.: 'Self-orientation of directional antennas, assisted by mobile robots, for receiving the best wireless signal strength', *Proceedings of Sensors Applications Symposium (SAS)*, 2012, pp. 1–6
20 Yan, J., Wan, Y., Fu, S., J., X., Li, S., Lu, K.: 'Rssi-based decentralized control for robust long-distance aerial networks using directional antennas', *IET Control Theory and Applications*, 2016, **11**, (11), pp. 1838–1847
21 BeagleBone: 'Beaglebone black', *BeagleBoardorg*, 2018, p. 1. Available from: `https://beagleboard.org/black`
22 QT: 'Software development framework', *QT | Cross-platform Software Development Framework*, 2018, p. 1. Available from: `https://www.qt.io/`
23 DJI: 'Matrice 100', *Matrice 100*, 2018, p. 1. Available from: `https://www.dji.com/matrice100`
24 Ubiquity: 'Nanostation loco m5', *Ubiquiti Networks*, 2018, p. 1. Available from: `https://www.ubnt.com/airmax/nanostationm/`
25 Huawei: 'Ws323 300mbps wireless range extender user guide', *Huawei Technologies*, 2018, p. 1
26 ROS: 'Robot operating system', *Rosorg*, 2018, p. 1. Available from: `http://www.ros.org/`
27 Rappaport, T.S., et al.: 'Wireless communications: principles and practice'. vol. 2. (PTR New Jersey, 1996)
28 Josefsson, L., Persson, P.: 'Conformal array antenna theory and design'. (John Wiley & Sons, 2006)
29 Kraus, J.D.: 'Antennas'. (McGraw-Hill Education, 1988)
30 Lewis, F.L., Vrabie, D.: 'Reinforcement learning and adaptive dynamic programming for feedback control', *IEEE circuits and systems magazine*, 2009, **9**, pp. 32–50
31 Lewis, F.L., Vrabie, D., Vamvoudakis, K.G.: 'Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers', *IEEE Control Systems*, 2012, **32**, pp. 76–105
32 Gil, A., Segura, J., Temme, N.M.: 'Numerical methods for special functions'. (Siam, 2007)