



# Building and Simulating Multi-Dimensional Drone Topologies

John Wensowitch, Mahmoud Badi, Dinesh Rajan, and Joseph Camp  
 Southern Methodist University  
 {jwensowitch,mbadi,rajand,camp}@smu.edu

## Abstract

The next wave of drone applications is moving from repeatable, single-drone activities such as evaluating propagation environments to team-based, multi-drone objectives such as drone-based emergency services. In parallel, testbeds have sought to evaluate emerging concepts such as highly-directional and distributed wireless communications. However, there is a lack of intersection between the two works to characterize the impact of the drone body, antenna placement, swarm topologies, and multi-dimensional connectivity needs that require in-flight experimentation with a surrounding testbed infrastructure. In this work, we design a Multi-Dimensional Drone Communications Infrastructure (MuDDI) to capture complex spatial wireless channel relationships that drone links experience as applications scale from single-drone to swarm-level networks within a shared three-dimensional space. Driven by the challenges of outdoor experimentation, we identify the need for a highly-controlled indoor environment where external factors can be mitigated. To do so, we first build an open-source drone platform to provide programmable control with visibility into the internal flight control system and sensors enabling specialized coordination and accurate repeatable positioning within the isolated environment. We then design a wireless data acquisition system and integrate distributed software defined radios (SDRs) in order to inspect multi-dimensional wireless behavior from the surrounding area. We achieve and demonstrate the value of measurement perspectives from diverse altitudes and spatial locations with the same notion of time. Finally, we demonstrate how multi-dimensional models from experimental measurements can be implemented to simulate multi-drone networks on a practical scale.

## ACM Reference Format:

John Wensowitch, Mahmoud Badi, Dinesh Rajan, and Joseph Camp. 2020. Building and Simulating Multi-Dimensional Drone Topologies. In *23rd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM '20), November 16–20, 2020, Alicante, Spain*. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3416010.3423235>

## 1 Introduction

The unique perspective that drones provide make them an attractive tool for numerous commercial applications. With over 436k commercial drones registered with the Federal Aviation Administration (FAA) as of February 2020 [1] and with the current projection for number of commercial registrations to be 2 to 3 years early [2],

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

*MSWiM '20, November 16–20, 2020, Alicante, Spain*

© 2020 Association for Computing Machinery.

ACM ISBN 978-1-4503-8117-8/20/11...\$15.00

<https://doi.org/10.1145/3416010.3423235>

many industries are quickly adopting drones for their operations. As drones move from single-platform, on-demand use cases to those with cooperative networks of nodes, it is important to understand the spatially-distributed challenges these wireless links face. Furthermore, due to the ad hoc nature of drone nodes and the intricacy of next-generation, multi-antenna wireless protocols, it is imperative to fully characterize these complex wireless channels across spatial distributions in the horizontal and vertical dimensions.

Several works use drones for measuring cellular coverage and interference [3–5], servicing Internet of Things (IoT) devices [6], and testbeds for developing enhanced vehicular control algorithms [7–9]. Others characterize the multi-dimensional propagation environment from a drone's perspective as well as the impact of antenna orientation in outdoor environments [10, 11]. There also exist wireless testbeds investigating next-generation wireless technologies and distributed networks for a variety of applications [12–15] as well as unique methods for distributed clocking [16]. We have used an anechoic chamber to characterize the impact of placing an antenna on a drone and rotating it such that a radiation pattern can be built over time from a single receiver [17]. However, these works do not allow the simultaneous reception from multiple perspectives across multiple azimuth and elevation angles for drone communications. For example, since various forms of directional transmissions can extend the range and corresponding size of a drone network, characterizing beam patterns transmitted from each drone or even from a set of antennas that are distributed across a swarm would be extremely challenging, if not impossible, in an anechoic chamber.

To focus on the effects that are localized to the drone, we draw on the challenges we experienced with outdoor drone-based measurements in order to make informed design decisions. However, outdoor experimentation alone has a number of issues such as the lack of control of external factors that lead to unrepeatable effects and an inability to set up wireless measurement nodes to encompass a flyable space in an ongoing manner due to weather. Hence, in this work, we design and build a Multi-Dimensional Drone Communications Infrastructure (MuDDI) to understand the spatial relationships that wireless drone links encounter. To carry out this vision, we address the following challenges with the described solutions:

*System-Level Integration of Programmable and Observable Subsystems:* First, we describe in detail the drone-flight-enabled wireless system and the experience that led to the specific indoor design. We explain the system-level integration of the infrastructure's supporting subsystems. We also describe the approach to addressing the challenges that an indoor system presents. While carrying out such an experiment, we additionally need to provide full observability of the drone platform and SDR network to facilitate a full view of the effects experienced by the drone.

*Sending Drones to Fixed Positions for Fixed Time Durations:* Without access to GPS, we seek to instruct a drone to fly to a certain indoor location and hold the position with high precision. As the

precision level increases, the swarm scale can correspondingly increase. To do so, we give a detailed breakdown of the selected open-source drone platform that enables accurate control and coordination within the system. We show that proprietary solutions lack built-in solutions for highly-controlled positioning in an indoor environment and describe the integration of our open-source drone build with an 8-anchor indoor localization system.

*Phase/Clock Alignment across 72 Spatially-Distributed RF Chains:* Lack of GPS also impacts state-of-the-art solutions for controlling large-scale SDR networks. In addition, we have found even within the same SDR, there are phase alignment issues across pairs of RF chains. We analyze these challenges faced with the distributed coordination. Namely, we experimentally show the phase alignment issues inherent across SDR boards and clocking issues when attempting to synchronize across tens of meters. We also provide data-driven technical solutions to these observed issues.

*Concurrent Wireless Capture from Various Horizontal and Vertical Positions:* To enable the greatest reach of our drone testbed to measure such effects as multipath or such applications as distributed beamforming, an extremely high sampling rate must be carried out at each SDR. When 10s of SDRs are performing this measurement concurrently, a simple combining of these data rates exceeds 2 Gigasamples/second (GS/s), which can overwhelm high performance servers. Hence, we develop a distributed-server solution and evaluate how these data capture rates can be maximized. We then show the degree to which synchronization can occur with first programmatic control and then post processing. Lastly, we perform a highly-controlled experiment to demonstrate the unique capability of the infrastructure.

*Modeling Drone-Based Communications for Multi-Drone Networks.* Finally, simulation-based modeling of experimentally measured drone-based effects is demonstrated. As potential applications require an increasing number of drones across various three-dimensional positions and orientations, a programmable and scalable environment enables complex scenarios to be evaluated. We demonstrate how drone induced effects can be accurately accounted for in a network simulation tool. Lastly, we highlight the ability of the drone-flight-enabled infrastructure to further validate developed models by isolating increasingly complex drone specific characteristics for multi-drone networks.

## 2 Drone-Flight-Enabled Chamber

The goal of our drone-flight-enabled isolation chamber is to ultimately support a swarm of drones to fly at specified locations, whether fixed or mobile, for tens of minutes with a surrounding wireless infrastructure to monitor the wireless activity of that drone swarm. The extent to which the drone network can scale in the infrastructure heavily depends on the degree to which drones can be trusted to reliably hold position and the size of the space in which the flights can take place. The indoor drone-flight-enabled wireless isolation chamber is housed in an indoor facility that allows approximately 20x20 m of flyable space. The minimum height within the flyable space is 5 m along the perimeter of the flyable space, but the pitched roof along the center of the building allows a maximum height of 7 m. Surrounding this flyable space, safety netting is installed for protection.

In an indoor environment, accurate positioning is challenging, especially with a metal roof, as is the case in our facility. In fact, in Section 3, we show that the interpretation of distances using an off-the-shelf drone controller from outdoor GPS in an unobstructed environment is so coarse that multiple drones flying inside a 20x20 m flyable space would be problematic. In addition, the drone platform needs to be able to interpret position from a positioning system and be programmatic so that it can receive directives to move to a given position for a given amount of time. Both aspects have been elusive in a widely-used off-the-shelf system that we have used extensively for outdoor experimentation, motivating our custom design of an open-source drone platform.

Isolating and inspecting a single wireless event requires all 72 RF chains that are dispersed over a large three-dimensional cube to be temporally synchronized. In Section 4.1, we articulate the challenges of building such a network and evaluate the timing alignment of that network with a highly-controlled experiment in Section 5. Moving beyond time, a secondary issue of phase alignment across even two boards that are housed within a single SDR has emerged both for using even parts of this very large antenna array as a multi-input, multi-output (MIMO) transceiver, or scaling up our drone-based antennas on a single body or across multiple drone platforms. Section 4.2 elaborates on the phase alignment issue since it is relevant in this SDR network.

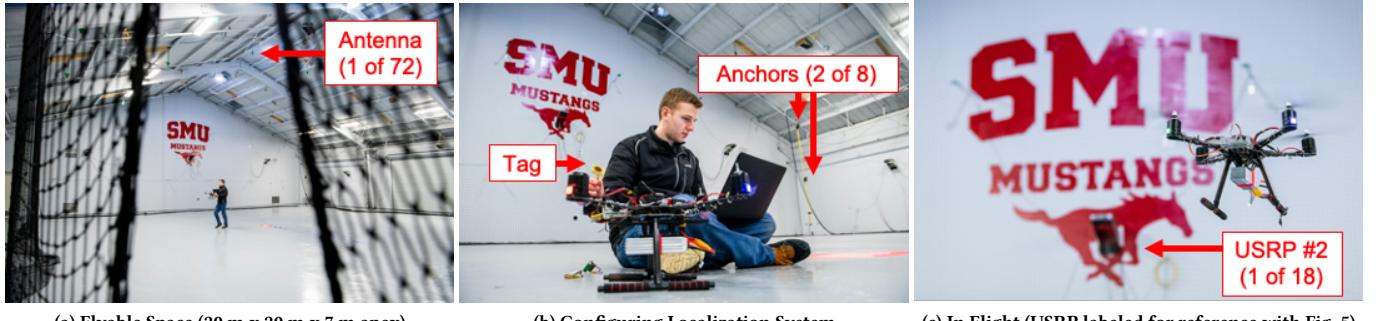
A final issue that we experienced in designing the drone-flight-enabled isolation chamber has to do with the data sampling rate desired at each of the RF chains. Next-generation MIMO applications require an extended bandwidth [18–20]. To enable multi-path and MIMO phase-level analysis, each RF chain needs a sampling rate of at least 30 MS/s. With 72 RF chains, we need to record over 2.1 GS/s from all SDR positions. We explore the tradeoffs of distributed logging per SDR versus centralized logging and real-time versus post-experiment processing. In Section 4.3, we describe the details behind a surprising finding that a real-time, centralized approach greatly outperforms a distributed, off-line approach, which is due primarily to the SDR design.

In summary, the infrastructure relies on the following interacting subsystems: (i.) an indoor localization system due to the lack of GPS, (ii.) an open-source drone platform that directly interfaces with the localization system, (iii.) eighteen SDRs that are distributed along the walls and ceiling, which are each attached to four log-periodic antennas pointed toward the center of the facility and connected to each RF chain, (iv.) a five-server system for distributed data acquisition, and (v.) three dedicated SDR-to-server cabling runs for clocking (fiber), data logging (fiber), and control (Ethernet). The overall system infrastructure is illustrated as a block diagram in Fig. 2. Additionally, a 2D layout of the flyable space is shown in Fig. 5<sup>1</sup>, pictured with an overhead view where the walls are laid down on the outside of the flyable space, and each SDR is numbered with a surrounding square.

## 3 Open-Source Drone Architecture

As mentioned previously, we have extensive experience working with off-the-shelf proprietary platforms in outdoor settings. However, to accomplish seamless integration into our infrastructure we need more control over the software architecture and design

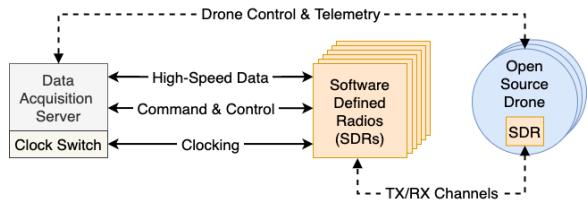
<sup>1</sup>To save space, we co-locate the floor plan and results from Section 5.



(a) Flyable Space (20 m x 20 m x 7 m apex). (b) Configuring Localization System. (c) In Flight (USRP labeled for reference with Fig. 5).

**Figure 1: Multi-Dimensional Drone Communications Infrastructure (MuDDI).**

as well as full visibility into the drone's flight control system. All of our efforts have been focused on multi-rotor drone platforms, more specifically quadcopters, that provide the ability to hover. This unique aspect of flight offers an unparalleled perspective for wireless applications. Designing a custom open-source drone platform with this capability took careful consideration of not only the flight dynamics and load capacity, but in the selection of open-source hardware and software to achieve our vision.



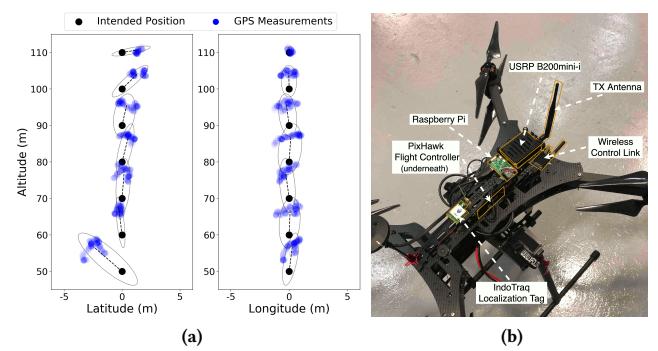
**Figure 2: High-level overview of drone-flight-enabled wireless isolation chamber infrastructure.**

Even commercially-engineered drone platforms from established companies such as DJI suffer from errors in accuracy when attempting to hold a stable hovering position. According to DJI's specifications [21], their common Matrice 100 quadcopter platform's flight controller can have errors of  $\pm 0.5$  m in the vertical plane and  $\pm 2.5$  m in the horizontal plane when maintaining a hover under ideal flight conditions. These errors are only made worse in an unpredictable outdoor environment where several other factors can affect flight controller performance and accuracy. Factors such as wind, GPS error, and atmospheric fluctuations in temperature and air pressure can induce unintended mobility and have a direct effect on the specified position and therefore the prediction and quality of a wireless channel.

For comparison, we demonstrate the accuracy of the popular DJI Matrice 100, a commercial load carrying drone based on DJI's N1 flight controller. Fig. 3a illustrates the three-dimensional error experienced by the drone's intended hover position. There are a total of seven intended hover locations that are marked with a bold black point. These points indicate the position instructed to the flight controller. The measured GPS locations are indicated by the smaller blue points and are reported using the USRP E312 as an external GPS receiver. The bounding circle indicates the intended position's corresponding measurements, while the dotted line indicates the distance from the intended point to the average of the received GPS measurements for that location. An external GPS receiver was used

to separate the discrepancy that the drone's flight controller sensors experienced from the experimental measurements. From this outdoor experiment, we see errors of up to 3.57 m in the latitude axis, 1.53 m in the longitude axis, and 9.49 m in the vertical axis, or altitude, from the intended position when attempting to hover in a programmatically-specified location. Due to these DJI limitations for this indoor application, we began working with commercially-available, open-source platforms. We first used a 3DR Solo, which is based on the widely-used, open-source ArduPilot flight controller software and PixHawk Cube hardware. This platform supports a payload capacity of 700 g and is meant for more consumer-level hobby and photography applications. This drone was outfitted with the lightweight (24 g) Ettus USRP B200mini-i SDR for the wireless communication link and connected to a Raspberry Pi 3 Model B+ (RPi) for on-board processing. The RPi bridged the computing gap between the SDR and the flight controller over a serial data connection. Using the wireless SDR link, the RPi communicated received control messages to the flight controller and allowed flight sensor data to be transmitted to the ground station. The full-duplex capability of the SDR provided a frequency-division-duplex link that was needed to implement our custom communication protocol. The uplink channel was dedicated to critical drone control messages, and the downlink channel provided sensor feedback to the ground station. This separation on two channels enabled a concurrent functional SDR uplink to the drone for programmatic flight commands from the ground station and downlink sensor feedback.

Following our experience with the 3DR Solo, we continued using ArduPilot software, the RPi as the companion computer, and PixHawk hardware for the flight controller on our subsequent drone



**Figure 3: Latitude, longitude, and altitude error from hover (a), and Open-source Drone Platform (b).**

builds. Compared to the 3DR Solo, our current custom drone platform has a much larger lift capacity. This increased capacity allows us to support multiple radio platforms that can be used for wireless experimentation (relevant for the phase offset alignment discussion in Section 4.2) in addition to a separate wireless link for drone control. All of our custom drone builds follow the basic design presented in Fig. 4, with the exception of the indoor localization that was recently added for precise positioning in MuDDI.

There exist various options for establishing a control link to the drone with varying degrees of reliability. Common control options include a manual remote control or a pre-programmed GPS way-point mission, which limits dynamic control and requires accurate GPS. Another option explored was using a standard WiFi link to the companion computer, but reliability and latency limited the performance. Furthermore, these technologies are limited to their designed frequency bands (900 MHz or 2.4/5 GHz), restricting controlled wireless experiments in these bands of interest. Hence, the SDR-based control link is an attractive option that enables dynamic frequency selection and the implementation of a custom control message format. However, designing a reliable, low latency protocol for an indoor environment with the limited SDR resources has proved challenging.

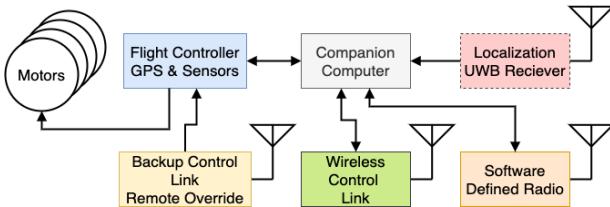


Figure 4: Block diagram of open-source platform.

With the accuracy and precision of positioning within the three-dimensional (3D) space being so important for repeatability and control, indoor localization was needed. In order to achieve a detailed level of positioning within our indoor flight space, we installed a 3D position tracking system by IndoTraq. To do so, we secured tripods to the ceiling at each of the four corners of the flyable space to hold two anchors each at heights of 2 m and 4 m for a total of 8 anchors. This system uses ultra-wide band wireless technology in addition to inertial-based tracking to provide sub-millimeter precision with update rates as high as 150 Hz while also providing localization throughout the entire flyable space. In an 8 anchor system, accuracy in all three dimensions is within 16 mm of the intended position [22]. The ultra-wide band technology operates at a frequency of 6.5 GHz, which is outside the operable frequency range of our SDRs and therefore eliminates the risk of interference in wireless measurements. A lightweight sensor tag (7 g) is integrated on our custom drone platform using a serial connection to the Raspberry Pi companion computer, and the calculated position is fed to the PixHawk flight controller in order to provide 3D positioning and tracking. Then, using the wireless control link, specified locations within the flight space can be programmatically and dynamically communicated to the drone. This indoor localization solution along with the isolated indoor chamber addresses the previous GPS precision inaccuracies and atmospheric factors by not relying on potentially-faulty flight controller sensors.

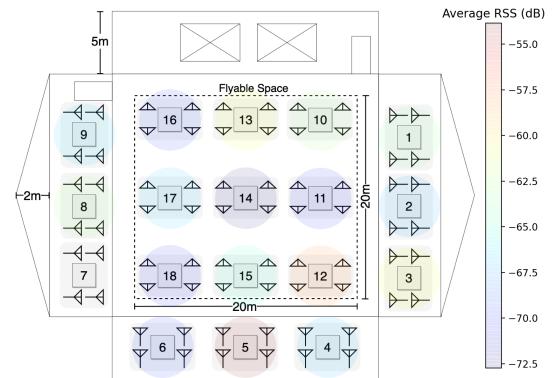


Figure 5: Layout of distributed N310s and antennas, average RSS highlighted from measurement results.

## 4 Software Defined Radio Network for Distributed Data Acquisition

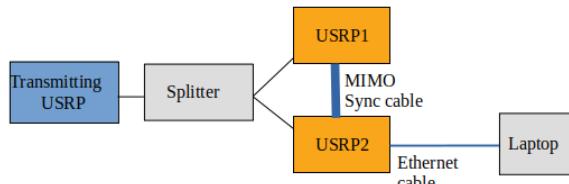
We now describe challenges with the SDR network on the ceiling and walls around the flyable space: (i.) clocking over long distances, (ii.) phase offsets even across boards in the same SDR, and (iii.) data logging from 72 RF chains.

### 4.1 Clocking Issues with Distributed SDRs

The SDR platform used throughout the facility is the Ettus USRP N310. This SDR is capable of simultaneously providing four full duplex channels (4 TX / 4 RX), each with a maximum instantaneous bandwidth of 100 MHz. It provides an extended frequency range of 10 MHz - 6 GHz and a sample rate of up to 153.6 MS/s. The N310 is a larger platform measuring 35.71 cm × 21.11 cm × 4.37 cm and weighing 3.13 kg, making it a challenge to fly on the drone. The SDRs that we have carried on the drone platform include the USRP B200mini-i and the USRP E312, weighing 24 g and 446 g, respectively. Both offer a tunable frequency range of 70 MHz - 6 GHz and provide up to 56 MHz of instantaneous bandwidth. The E312 offers two full duplex channels, while the B200mini-i offers only one. The E312 possesses an embedded Linux system running on an ARM Cortex A9 chip enabling stand-alone operation, where the B200mini-i relies on a USB connection to a host. This makes the B200mini-i more suited for integrating drone control over an SDR link with a companion computer, while the E312 is more effective for collecting raw measurements or running a stand-alone protocol.

Each N310 has four LP0965 log-periodic PCB antennas connected to the RF front end that is operational over the entire operating carrier frequency range of the SDR. The 6-dBi antennas either point directly towards the floor from the ceiling mounts or directly towards the opposing wall from the wall mounts. Each antenna connects to the TX/RX port allowing for time-division duplexing. With four antennas at each SDR, a total of 72 simultaneous channels have the ability to transmit, receive, or any combination of the two. Each of the eighteen SDRs have been strategically located to cover the flyable space with a 3-m spacing between adjacent antennas. Fig. 5 shows the radio and antenna layout.

With so many radios operating together in a distributed fashion, a clocking solution was needed to provide synchronous operation. The first option explored was Ettus' OctoClock solution that provides synchronization for up to eight devices per OctoClock device.



**Figure 6: Experiment setup for phase offset testing.**

However, this solution is operated over copper connections and required a GPS input for a disciplined clock source. These factors proved challenging for several reasons. First, the max distance from the central clock source would be approximately 30 m away when the cables were routed. This is not practical for copper-based connections as these distances would experience significant cable losses as well as introduce unnecessary drift and noise in the clock pulses, resulting in unacceptable errors in synchronization. Secondly, the indoor environment lacked a reliable GPS signal which would further degrade the quality of the clock source. The clocking method that we chose to incorporate in our system was the White Rabbit Network Switch from Seven Solutions. This solution provided sub-nanosecond time accuracy over fiber connection for up to eighteen devices on one network switch. Since it operates over a fiber link, it can support multiple-km distances, easily supporting our needs. It also didn't require a disciplined clock and still offers a clocking accuracy to within 1 nanosecond [23]. Using the White Rabbit clock, we ran single-mode fiber cabling from a central control point to each of the eighteen radios in order to provide accurate and distributed clocking, which we experimentally evaluate in Section 5.

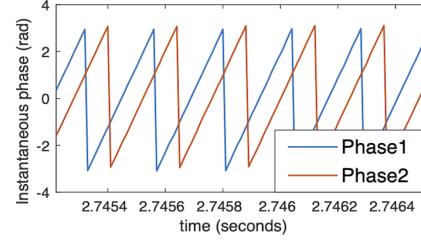
## 4.2 Phase Errors Across SDR Boards

Even with time synchronization and shared clocking among all the distributed USRPs, there still exists a random phase generated by each USRP's local oscillator. In fact, in the case of N310 with two daughterboards (2 RX/TX channels per board), there will be an intra-phase offset in addition to the inter-phase offset that usually exists between different USRPs. To the best of our knowledge, aside from the manufacturer's application notes regarding this problem, there is very little information available in literature regarding how this phase error in USRPs behaves across time and multiple experiments. For example, in [24], while carrier frequency offset (CFO) and phase errors are studied, the sudden drifts in time are not discussed. Even in [25], researchers only study 1.25 seconds of recorded phase values, which we will show is not sufficient to capture the temporal nature of the phase error since sudden phase drifts can occur due to temperature variations or frequency divider ambiguities. To characterize this phase offset issue between two boards, we conduct the experiment shown in Fig. 6.

Following the manufacturer's instructions [26], we send a 1-KHz sine wave at a sampling rate of 100 kS/s to two synchronized N210 USRPs through a splitter with two matching-length cables. The two USRPs are synchronized using a MIMO cable that shares the 10 MHz/PPS signals of one USRP (master) with the other (slave). The 10 MHz clock and PPS signal are required for time and frequency synchronization. However, there is still a random phase generated by each USRP's independent local oscillator. We aim to characterize this random phase offset, necessary for phase coherent applications such as distributed beamforming. The phase difference between the two received signals at the two USRPs is calculated using the

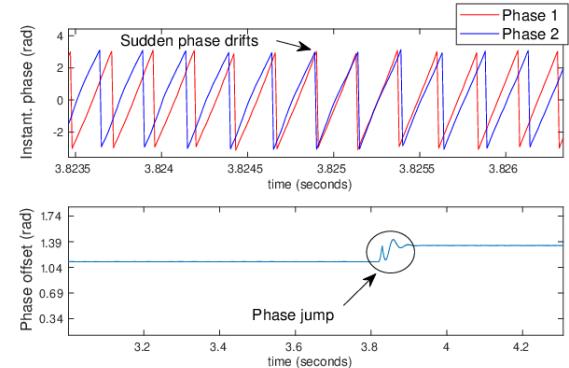
unwrapping method. Namely, we unwrap each received phase over time and calculate the difference between the two unwrapped angles. We conduct 11 experiments (10 for 10 s, 1 for 120 s) for 3 carrier frequencies of 500 MHz, 2.4 GHz, and 3.5 GHz, but we show results for only 3.5 GHz due to limited space. All experiments are conducted at room temperature. We find the following:

(i.) With each experiment there exists a random phase error. This random phase error is, for all trials that have no sudden phase drifts, approximately constant over time with a standard deviation that does not exceed 0.03 radians across all trials. An example of this phase difference is shown in Fig. 7.



**Figure 7: Synchronized instantaneous phase for nodes**

(ii.) There exists sudden phase drifts over time, causing jumps in the phase offset value. These phase drifts, and consequently, phase jumps, occur within a 1-s interval with a maximum number of 2 times per experiment. This measurement tells us that when using distributed USRPs, even when synchronized, there is a need for a continuous phase calibration and correction procedure in order to capture these random phase jumps. In other words, a one-time calibration before experimenting with distributed USRPs is not enough: a finding that can impact how distributed SDRs are designed for phase-coherent applications. An example of a sudden phase drift of 0.22 radians and the resulting jump in phase error is shown in Fig. 8.



**Figure 8: Sudden phase drifts causing jumps in phase difference between synchronized USRPs.**

(iii.) To address the temporal nature of this phase offset, we use a moving time window of a 2-s intervals and find that, if the experiment is free of the sudden phase drifts discussed above, the mean and standard deviation of the phase offset are approximately constant over the different time periods. Therefore, a mean over the 10 s (or 120 s in the long experiment) is representative of this phase offset and can be used in a one-shot calibration procedure where we compensate for the phase error using a reference phase. However, as mentioned above, this is not always the case. If sudden phase drifts

Trial	1	2	3	4	5	6	7	8	9	10	11 (120 s)
Mean (rad)	1.678	2.228	2.007	2.114	1.674	12.028	1.771	-0.0347	-0.1850	1.890	2.494
Standard Deviation (rad)	0.0217	0.0281	0.0264	0.0271	0.0198	0.1096	0.0199	0.0201	0.0203	0.1491	0.1135

**Table 1: Phase offset between synchronized distributed USRPs.**

occur, the mean and standard deviation obtained over the whole experiment duration are no longer representative due to sudden jumps in value. Therefore, a continuous calibration procedure is needed where the phase values are checked every 1 s as a worst case scenario. A summary of the mean phase offset and its standard deviation in radians is presented in Table 1. Notice how the standard deviation of experiments 6, 10, and 11 is greater due to the sudden phase drifts that occurred in those trials.

### 4.3 Commanding/Logging 72 RF Chains

Another critical challenge faced with operating eighteen SDRs in such a distributed fashion is providing command and control to each radio. The other challenge that logically followed was collecting, generating, and processing the large amount of data that would either need to be created or captured at each radio. First, in order to provide remote command and control, a wired 1 Gbps Ethernet-based connection was established to each of the radios. This allowed for remote configuration from a central point but didn't provide a method for collecting, processing, or storing measurement data. To address this we took a methodical approach in assessing different techniques for handling this challenge.

We first explored processing samples locally at each N310. However, each radio was limited to the local ARM Cortex-A9 processor and used a flash-based microSD card for data storage. Using a single N310 to benchmark a radio's performance, first the Unix command-line utility 'dd' was used to test the disk write speed in MB/s for several different microSD cards from various manufacturers. Next, based on the max disk performance, a theoretical sample rate was calculated to predict the max achievable sample rate. This calculation was based on the fact that each sample written to file is a complex float producing a total of 8 bytes per sample or 4 bytes per I and Q, respectively. Finally, the true measurement capability was assessed by using a benchmark utility provided by Ettus and slowly backing off the predicted sample rates until there was no longer data overflow, meaning loss of samples due to the lack of buffer space. Each microSD card's speed class, capacity, quoted write speed, 'dd' write speed results, calculated theoretical max rate, and benchmark utility test results were recorded. The results from these tests are presented in Table 2. From these results, we show that the max sample rate achieved was approximately 1.5 MS/s, greatly underutilizing the N310's max capability of 153.6 MS/s.

Next, we shifted from locally-processing samples, to attempting to send and receive samples using a networked mode. In order to achieve this process and handle the large amount of data associated with higher sample rates, we identified the need for a high performance computing platform. To better understand the computing needs and data rate budget, we started with one server grade machine to test how many radios we could support on a single machine before scaling to a multi-server setup to service all eighteen radios. We used a dedicated quad-port SFP+ network interface card (NIC) operating on Peripheral Component Interconnect express Revision 3 (PCIe Rev. 3) to provide a direct link from the N310 to the server

in order to get the best data rate and network performance without router or switch limitations.

The first cabling method tested from the N310 to the server was CAT 6 RJ45 Ethernet, supporting 10 Gbps on a SFP+ adapter at both the radio and NIC. However, when testing the max distance cable length of 30 m, the link did not support the full 10 Gbps speed and reverted to a 1 Gbps link. Since this issue would bottleneck the data link budget, we then moved to using a full duplex multi-mode fiber connection over a supporting SFP+ adapter at both the radio and NIC. With the full 10 Gbps speed in both the uplink and downlink confirmed, several tests using a network-mode benchmark utility allowed us to gauge how many radios we could support using one machine. Using up to three quad-port NICs on the single server-grade test machine, we benchmarked the performance for a total of 4, 6, and 9 network-mode links with the radio front end connection configuration, set receiver (RX) sample rate, and set transmitter (TX) sample rate noted. Then, each test was performed for 10 s.

The results for the various tested combinations of receiving RX and generating TX sample rates are shown in Table 3. The front end connections refer to the number of front end connections per radio, so 4-RX and 4-TX across four N310s signifies a total of 16 channels of both RX and TX. The tests that completed without an underflow or overflow in the network buffers are indicated with a pass (P), where those that experienced at least one buffer issue are indicated with a fail (F). These front end combinations were selected to assess the best and worst case scenarios for different configurations. From these results, we conclude that connecting four radios per machine in the server design would provide the best network performance for processing samples and still achieve near the maximum capability of the N310 hardware.

Therefore, a total of five servers were custom built to match the test machine and provide dedicated connectivity to each of the eighteen SDRs. Each server was outfitted with one quad-port NIC, 2 terabytes (TB) of high-speed solid-state storage (2 x 1 TB in RAID 0) with a maximum write speed of 3.3 GB/s, and 10 TB of hard disk space that allows offloading from high-speed storage to enable various combinations of collection times and sample rates.

$$\text{Data Rate}(DR) = \text{Sample Rate}(SR) \times 8 B \times \# \text{ Channels}(C) \quad (1)$$

$$\text{Collection Time}(CT) = \frac{\text{Disk Space}}{\text{Data Rate}} \quad (2)$$

Equations 1 & 2 demonstrate the theoretical maximum collection time based on the sample rate and number of channels. Recall, each server has a maximum of four connected N310s and therefore, a maximum of 16 channels per server. Using the high-speed 2 TB storage and sample rates of 30 MS/s across all 16 RX channels for one server results in a maximum collection time of 520.8 s.

### 5 Multi-Dimensional Data Capture

In order to demonstrate the scale and level of control at which we can perform experiments, we provide a simple experimental setup for capturing received samples from a transmitting drone. First, to achieve a distributed level of control, the servers institute a parallel

Card	Class	Capacity (GB)	Write (MB/s)	'dd' (MB/s)	Max Rate (MS/s)	Tested Rate (MS/s)
Samsung Evo	10, U3	512	90	17.5	2.1875	1
Samsung Evo	10, U3	256	90	15.6	1.95	1.201923
SanDisk Extreme	U3, V30	400	90	17	2.125	1.506024
SanDisk Extreme	U3, V30	128	90	17.2	2.15	1.404494
Lexar Professional	U3, V90	128	150	10.9	1.3625	0.5

Table 2: Write and sample rate performance of microSD cards.

Front End	RX (MS/s)	TX (MS/s)	Pass/Fail
Four - N310			
1-RX	125	N/A	P
1-RX	125	N/A	P
1-RX 1-TX	125	125	P
1-RX 1-TX	125	125	P
4-RX 4-TX	62.5	62.5	P
4-RX 4-TX	62.5	62.5	P
4-RX 4-TX	125	62.5	F
Six - N310			
1-RX	62.5	N/A	P
1-RX	125	N/A	P
4-RX	31.25	N/A	P
4-RX	62.5	N/A	F
4-RX	125	N/A	F
1-RX 1-TX	62.5	6.25	P
1-RX 1-TX	62.5	12.5	F
4-RX 4-TX	31.25	1.25	P
4-RX 4-TX	62.5	1.25	F
4-RX 4-TX	12.5	6.25	F
4-RX 4-TX	31.25	6.25	F
Nine - N310			
1-RX	20.83	N/A	P
1-RX	31.25	N/A	P
1-RX	62.5	N/A	F
4-RX 4-TX	1.25	1.25	P
4-RX 4-TX	31.25	31.25	F

Table 3: Achievable sample rates in networked mode based on number of SDRs connected to a single server.

control scheme that issues command messages over the Ethernet network and direct each USRP to their respective server over fiber to start a measurement collection. Using Ettus’ USRP Hardware Driver (UHD) software API, a simple write-to-file application is created to sample 1.25 MHz of bandwidth at a center frequency of 5.15 GHz. For this demonstration, we capture one RX channel across all eighteen SDRs simultaneously. In order to isolate the effect of receiving from each location, the drone is placed on a fixed platform at the center of the flyable space directly below SDR #14 in Fig. 5 at a height of 3.5 m. The drone has a vertically-oriented, omni-directional TX antenna (pictured in Fig. 3b) on the side of the drone nearest SDR #2. The transmitter is a USRP B200mini-i SDR on the drone with a narrow-band sine wave at a center frequency of 5.15 GHz in a periodic on-off pattern and a cycle frequency 0.5 Hz. Then, one receiver chain from each of the eighteen N310s simultaneously samples at a rate of 1.25 MS/s for a total of 30 seconds. The IQ samples are stored from each SDR on their respective server and are later processed for the received signal strength (RSS) in dB.

Fig. 9 shows a one-second portion of the transmitter-on cycle for a temporal representation of the RSS. In order to better visualize the general trend, a moving average was taken over a windows size of 3000 samples. The top graph shows the best synchronization that can be achieved with programmatic control alignment of sampled data by using a timestamp from the start of the collection. While the bottom graph shows the post-processed alignment, based on the rising edge of the first cycle. There was a maximum of 26,907 samples, or 21.53 milliseconds, between the first and last rising edge using the programmatic control method of synchronization. After post-processing based on the rising edge, both the rising and falling edges were aligned to within 800 ns.

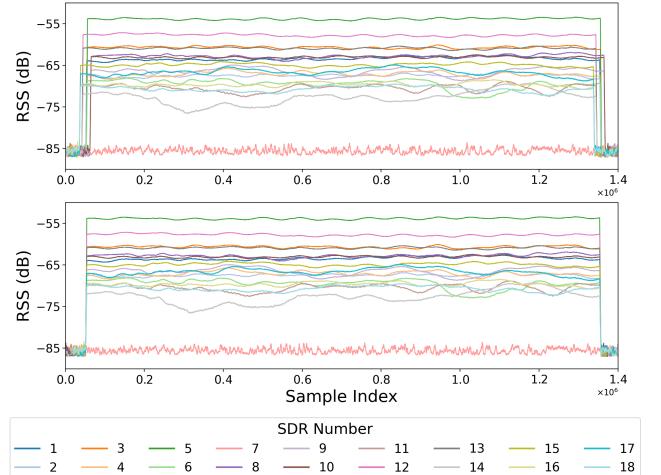


Figure 9: Average RSSI with programmatic (top) and post-processed (bottom) synchronization.

By inspecting the RSS values in Fig. 9, we see a maximum of a 23.12 dB difference in the RSS from SDR #14 to SDR #5, across all eighteen spatially-distributed receivers. We can additionally refer to Fig. 5, which illustrates the overall average RSS experienced at each SDR in the form of a colored heat-map overlay. Both of these figures indicate that the drone experiences vastly different propagation effects from approximately equidistant receivers in this controlled environment. In particular, the receivers located above the drone are consistently lower in terms of their received signal levels as compared to the antennas on the walls, likely due to the transmitting, vertically-oriented omni-directional antenna, which has the lowest radiation pattern value at that elevation angle. Furthermore, there is diversity even among the walls, highlighting the potential impact the drone body itself is having on the *omni-directional* antenna mounted on the SDR #2 side of the drone. This further highlights the need to better understand the complex three-dimensional effects that drones experience, a key component that our unique drone-flight-enabled wireless isolation chamber enables.

## 6 Building a Drone-Based Simulation Environment

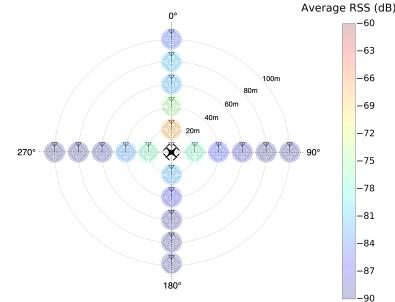
Our work towards a drone-based simulation environment focuses on modeling our experimental understanding and implementing those results in a computational setting. In particular, an emphasis is placed on emulating the physical layer characteristics that we have identified and accurately representing their impact on the wireless channel. Compared to standard wireless propagation and elevation models, our experimental results show an increased need for characterizing multi-dimensional drone specific aspects to properly represent practical drone-based communications. Aspects such as the relative three-dimensional location of drone nodes, the physical drone body, placement of antennas, and orientation of the antennas require several variations in standard models to effectively represent their induced effects. Properly capturing and representing these effects becomes increasingly important as the scale of the drone network increases. By accurately modeling the multi-dimensional characteristics, we can predict the complex communication channels among large networks of drones across diverse three-dimensional positions and altitudes.

Rather than programming a custom simulation environment from the ground up, we choose to use OMNeT++ with INET, a C++ based framework that provides several models that can easily be customized to represent the required elements of multi-dimensional drone-based communications. In particular, we need a programmable environment to simulate a complex network stack, implement several physical layer aspects, create various wireless interfaces, provide three-dimensional mobility, and assess radio and network-level performance. OMNeT++ with INET provides a sufficient starting point for all of these requirements.

In our initial simulation work, we demonstrate how experimental results from our outdoor measurement studies and indoor anechoic chamber experiments in our previous work [17] can be transferred to a simulation environment for the evaluation and validation of drone-based communications. In particular, we explore a two-dimensional scenario that takes into account the effect of the drone body as well as multiple antenna orientations and positions. It is important to note that to effectively target the drone induced effects, all of the described experiments utilize the same antenna across similar frequencies, eliminating any physical antenna dependent characteristics when comparing results. However, in the outdoor environment we lack the isolated control of several variables to completely separate undesirable effects from the intended experimental variables. Even in the isolated environment of the anechoic chamber, limitations prevent the characterization of altitude effects, in-flight effects, and the assessment of multiple drones at once.

We first focus on a simple two-dimensional representation of the relative angle between a transmitting drone to a receiving drone in an outdoor, line-of-sight environment. The general layout and results from this in-field experiment are represented in Fig. 10. An important aspect of this experiment is that both drones are fixed at the same elevation and have their forward-mounted antennas facing towards 0 degrees (fixed yaw) throughout this experiment, creating a unique scenario for assessing two-dimensional directional losses. Fig. 10 also illustrates the distances and angles that are measured with the receiving drone in the middle and the transmit drone

positions highlighted with the average received signal strength (RSS) experienced by the receiving drone along each axis creating angles of 0, 90, 180, and 270 degrees. From this figure, it is clear that the angle and distances in front of the receiving drone experience a higher RSS compared to the other measured locations.



**Figure 10: In-field two-dimensional experiment RX drone (center) with TX drone positions (surrounding).**

It is common to assume that in a fixed horizontal, or azimuth plane that the radiation pattern and therefore observed losses would be constant no matter the direction of the transmitter from the receiver. However, from our two-dimensional experiment it is clear that once the antenna was mounted on the drone and the transmit-receive angle is changed, the constant azimuth assumption is no longer valid. Using a standard path loss model, we observe that a constant azimuth assumption greatly underestimates the path loss experienced by the drone by up to 23.15 dB. To account for these losses, we represent the differences in path loss by adding a shadowing factor to a log-normal path loss model for each of the measured angles. This modified model allowed us to accurately predict and simulate the RSS for each measured angle. This modified model produces a maximum average prediction error of 4.10 dB in simulation for the angle of 180 degrees. When compared to the standard model error of up to 25.58 dB, we demonstrate an 83.9% reduction in error.

To simulate this experiment within OMNeT++, an ad hoc drone network is created to match the experimental setup and parameters. An existing wireless AdhocHost module is used to represent the transmitting and receiving drones. We then use the Ieee80211Scalar-RadioMedium module to simulate the wireless physical layer parameters. Within this module, an omni-directional dipole antenna is chosen to match the in-field experiment antenna characteristics. Next, an Ipv4NetworkConfigurator module is used to assign IP addresses and create the routing between network interfaces. Finally, to implement our modified path loss model, a custom module is created in C++ to capture the relative transmit-receive angle and calculate the corresponding path loss based on the added shadowing factor. This calculation uses a lookup table and is only valid for the four measured angles.

Equations 3 & 4 summarize the modified path loss model that is used for this calculation. In Equation 3, the transmit power and receive power are represented by  $P_t$  and  $P_r$ , respectively. Then,  $P_L(d_0)$  is the reference path loss calculation for a distance of 20 m,  $\alpha$  is the path loss exponent, and  $\xi_s$  is the zero-mean, normally distributed shadowing parameter with a standard deviation of 2. In Equation 4,  $PL_{UAV}$  represents our modified path loss value,

where  $\Gamma_\phi$  is the additional averaged shadowing parameter that was induced by the drone in the corresponding azimuth angle ( $\phi$ ). To produce an accurate calculation within the simulation, the same center frequency (2.5 GHz) and transmit power (6.2 dBm or 4.1687 mW) parameters are matched. Finally, the simulation is run at 1 m increments from a distance of 20 m to 100 m while the received power is logged at each location.

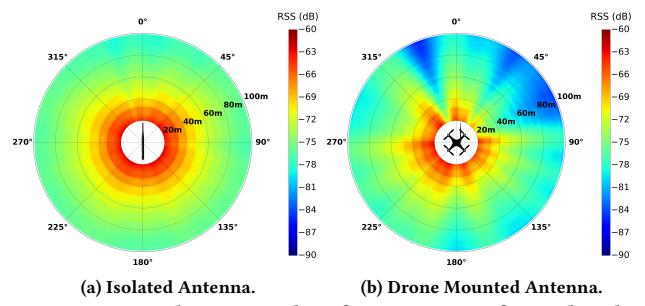
$$P_r = P_t - PL(d_o) - 10\alpha \log(d/d_o) + \xi_s \quad (3)$$

$$PL_{UAV} = PL_{log-distance} + \Gamma_\phi \quad (4)$$

The simulated results using our developed model for the angles of 0 and 180 degrees adds an additional shadowing factor ( $\Gamma_\phi$ ) of 9.17 dB and 21.45 dB, respectively. When compared to the in-field experiments where only five distances were used to develop the model, we show that on average, the same accuracy is achieved in distance increments of 1 m. While this two-dimensional path loss model highlights the substantial effect that the drone body had on predicting RSS, it did not capture all 360 degrees of yaw or the various complex antenna positions and orientations that we wish to simulate on a larger scale nor does it provide a means of separating the drone induced effects from the drone-based transmitter.

Once we identified that the drone body was a substantial source of attenuation, we shifted focus from a path loss based model to characterizing the attenuation using a directional gain based approach. A gain based approach to modeling the unique drone-based aspects allows assessment of how the directivity of the wireless signal is affected from the perspective of the antenna by the complex influence of the drone body as well as the various antenna positions and orientations. This approach also allows for the simulation environment to be easily extensible to various drone bodies and antenna configurations by simply characterizing a specific drone and antenna setup once and scaling to any physical network topology necessary. This shift in modeling is chosen because path loss is typically associated with characterizing the surrounding propagation environment and not the direct effect of the drone body or the characteristics induced on the communicating antennas. It also allows for the logical separation of transmitting and receiving elements and their induced effects when evaluating drone-to-drone networks with various antenna configurations. Since our outdoor experiments are performed at an altitude where unobstructed line-of-sight propagation is assumed, the concept of path loss does not completely capture the observed drone-based effects.

Following this realization, we utilize the anechoic chamber to characterize the antenna gain in great detail both on and off the drone in a highly isolated manner. Again, due to the limited capabilities of the chamber, we are only able to accurately describe the yaw dimension, or azimuth plane, of the antenna for a vertical antenna orientation. Within the anechoic chamber, we characterize the antenna alone versus mounted on the drone, as described in Section 2 of our previous work [17]. The controlled nature of the chamber enables us to capture a greater level of detail about each antenna configuration in the azimuth plane. Fig. 2 in [17] shows the setup within the anechoic chamber and the drone-mounted antenna position measured on the drone body, forward right. The drone is mounted on a rotating platform controlled by a stepper motor with a resolution of 1.8 degrees. This provides a total of 200 individual antenna gain measurements for each antenna configuration.



**Figure 11: Simulation results of gain pattern for isolated & drone mounted antenna.**

Using this highly-detailed characterization of the gain pattern experienced by the antenna, the same in-field experiment is recreated in the simulation environment. However, now we were able to provide a full representation of the azimuth plane both with and without the drone body for all 360 degrees of yaw and the same experimental distances. Within OMNeT, the same basic drone network setup is used as in the previous simulation. However, this time instead of a dipole antenna, the transmitting and receiving drones are configured with an interpolating antenna that models a directional antenna based on a sequence of angles and gain values. Using the 200 angles and gain values measured in the chamber, these antennas are set up to match the exact characterization for each antenna configuration. Again, the same transmit power and center frequency is matched, but this time a free-space path loss model is used rather than the modified log-normal path loss model. By leveraging this detailed understanding of the effective antenna gain patterns, we illustrate the impact that the drone body has within the simulation environment. Fig. 11 provides a heat map representation of the predicted RSS for both antenna configurations across the same distances as our in-field experiments. The color map key represents the expected RSS in dB experienced by the receiving drone at the center from each of the surrounding transmit locations with a 1 m resolution. As expected, the isolated antenna shows a gain pattern in the azimuth plane that is characteristic of a vertically-orientated omni-directional dipole antenna. Compared to the isolated antenna configuration pattern, the drone has a significant effect on the achievable gain of the antenna and therefore the RSS for different transmit-receive angles and distances.

One interesting observation from Fig. 11b is that the forward-facing angle opposite of the receiving antenna placement experiences substantial losses (dark-blue cone pattern) starting at 40 m centered at approximately 335 degrees. This observation is important when predicting the effective distance and channel quality as these losses will limit the potential range and/or achievable throughput in multi-drone networks.

At this point in our simulation work, we plan to perform several experiments within our indoor infrastructure to move towards a more controlled understanding of the unique three-dimensional aspects that we have identified thus far. In particular we plan to first focus on describing the elevation plane effects to provide a three-dimensional understanding of the drone body effect. The simultaneous multi-dimensional capture capability, as well as the highly-precise control of drone position and orientation, will enable us to fully characterize these unique effects from the perspective

of a drone-based receiver and transmitter. Furthermore, due to the dynamic and configurable nature of the SDRs, it is possible to quickly switch the surrounding fixed antennas to receiving or transmitting elements, effectively separating the undesired drone induced variables we faced in the outdoor experiments. These insights will enable the validation of multiple drone-based scenarios as well as the dynamic switching or movement of antenna positions and orientations providing an isolated characterization to bring greater levels of fidelity to the simulation environment for multi-dimensional multi-drone networks.

## 7 Related Work

There are several existing works that fall into two broad categories relating to our efforts. In the first category, drones have been used as tools within wireless research, such as assessing propagation [10, 19, 27], or improving path planning and control systems algorithms [3–5]. Other works have even used fixed wing drones to further explore how antenna type and orientation affected achievable throughput [11] or serviced IoT devices with groups of mobile drones [6]. MIT’s Indoor Multi-Vehicle Flight Testbed [7] coordinate several drones with localization for assessing control system performance, but wireless link characterization is not discussed. Lastly, simulation environments have tested coordination and control algorithms for drones with some indoor validation [8]. In the second category, wireless testbeds have been developed to evaluate various communication technologies. The PAWR Project [12] is helping to develop various emerging technologies such as massive MIMO and 5G connectivity. The Open Access Research Testbed for Next-Generation Wireless Networks (ORBIT) [15] provided a large scale wireless testbed for indoor reproducibility as well as an outdoor network testbed to test real-world scenarios. In AirShare [16] they proposed a novel method of sharing a reference clock to independent wireless nodes in a distributed manner for use in distributed MIMO and rate adaptation applications. In contrast to these efforts, we build a programmable testbed around a flyable space to inspect drone networks.

## 8 Acknowledgements

This work was supported in part by NSF grant CNS-1823304 and Air Force Office of Scientific Research grant FA9550-19-1-0375.

## 9 Conclusion

In this work, we described our experience and lessons learned from building a drone-flight-enabled wireless isolation chamber, which allows in-depth evaluation of the complex spatial relationships that are experienced in drone communications. To do so, we first built an open-source drone platform that allowed us to directly interface with an indoor localization and programmatically position drones for a given amount of time. Next, we designed a wireless data acquisition system with a family of SDRs over the outside of the flyable space in the testbed, which presented challenges for synchronization, phase alignment, and data logging. We demonstrated the culmination of this integration with a highly-controlled experiment from the same transmitter mounted on a drone and show an ability to synchronize and evaluate three-dimensional aspects in drone networks. Finally, we demonstrated how our experimental measurements can be leveraged to simulate multi-dimensional drone networks on a practical scale.

## References

- [1] UAS by the numbers. [https://www.faa.gov/uas/resources/by\\_the\\_numbers/](https://www.faa.gov/uas/resources/by_the_numbers/), Feb. 2020.
- [2] FAA aerospace forecast fiscal years 2019-2039. [https://www.faa.gov/data-research/aviation/aerospace\\_forecasts/media/FY2019-39\\_FAAG\\_Aerospace\\_Forecast.pdf](https://www.faa.gov/data-research/aviation/aerospace_forecasts/media/FY2019-39_FAAG_Aerospace_Forecast.pdf), 2019.
- [3] Vijaya Yajnanarayana, Y.-P. Eric Wang, Shiwei Gao, Siva D. Muruganathan, and Xingqin Lin. Interference mitigation methods for unmanned aerial vehicles served by cellular networks. *CoRR*, abs/1802.00223, 2018.
- [4] Drones and networks: Ensuring safe and secure operations, Dec. 2019.
- [5] Xingqin Lin, Richard Wiren, Sebastian Euler, Arvi Sadam, Helka-Liina Maattanen, Siva Muruganathan, Shiwei Gao, Y-P Eric Wang, Juhani Kauppi, Zhenhua Zou, et al. Mobile network-connected drones: Field trials, simulations, and design insights. *IEEE Vehicular Technology Magazine*, 14(3):115–125, 2019.
- [6] Mohammad Mozaffari, Walid Saad, Mehdi Bennis, and Merouane Debbah. Mobile internet of things: Can UAVs provide an energy-efficient mobile architecture?, Jul. 2016.
- [7] M. Valenti, B. Bethke, D. Dale, A. Frank, J. McGrew, S. Ahrens, J. P. How, and J. Vian. The MIT indoor multi-vehicle flight testbed. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 2758–2759, Apr. 2007.
- [8] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar. The GRASP multiple micro-UAV testbed. *IEEE Robotics Automation Magazine*, 17(3):56–65, Sep. 2010.
- [9] J. P. How, B. Behlirke, A. Frank, D. Dale, and J. Vian. Real-time indoor autonomous vehicle test environment. *IEEE Control Systems Magazine*, 28(2):51–64, Apr. 2008.
- [10] Evşen Yanmaz, Robert Kuschnig, and Christian Bettstetter. Achieving air-ground communications in 802.11 networks with three-dimensional aerial mobility. In *2013 Proceedings IEEE INFOCOM*, pages 120–124. IEEE, 2013.
- [11] C. Cheng, P. Hsiao, H. T. Kung, and D. Vlah. Performance measurement of 802.11a wireless links from uav to ground nodes with various antenna orientations. In *Proceedings of 15th International Conference on Computer Communications and Networks*, pages 303–308, Oct. 2006.
- [12] Platforms for Advanced Wireless Research. <https://advancedwireless.org/>, Mar. 2020.
- [13] R. Lim, F. Ferrari, M. Zimmerling, C. Walser, P. Sommer, and J. Beutel. Flocklab: A testbed for distributed, synchronized tracing and profiling of wireless embedded systems. In *2013 ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 153–165, Apr. 2013.
- [14] J. Vieira, S. Malkowsky, K. Nieman, Z. Miers, N. Kundargi, L. Liu, I. Wong, V. Öwall, O. Edfors, and F. Tufvesson. A flexible 100-antenna testbed for massive MIMO. In *2014 IEEE Globecom Workshops (GC Wkshps)*, pages 287–293, Dec. 2014.
- [15] D. Raychaudhuri, I. Seskar, M. Ott, S. Ganu, K. Ramachandran, H. Kreimo, R. Siracusano, H. Liu, and M. Singh. Overview of the ORBIT radio grid testbed for evaluation of next-generation wireless network protocols. In *IEEE Wireless Communications and Networking Conference, 2005*, volume 3, pages 1664–1669 Vol. 3, Mar. 2005.
- [16] Omid Abari, Hariharan Rahul, Dina Katabi, and Mondira Pant. Airshare: Distributed coherent transmission made seamless. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 1742–1750. IEEE, 2015.
- [17] M. Badi, J. Wensowitch, D. Rajan, and J. Camp. Experimental evaluation of antenna polarization and elevation effects on drone communications. In *Proceedings of the 22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 211–220, 2019.
- [18] M. Rice and M. Jensen. Multipath propagation for helicopter-to-ground mimo links. In *2011 - MILCOM 2011 Military Communications Conference*, pages 447–452, 2011.
- [19] T. J. Willink, C. C. Squires, G. W. K. Colman, and M. T. Muccio. Measurement and characterization of low-altitude air-to-ground mimo channels. *IEEE Transactions on Vehicular Technology*, 65(4):2637–2648, 2016.
- [20] T. Zhou, C. Tao, and L. Liu. LTE-assisted multi-link MIMO channel characterization for high-speed train communication systems. *IEEE Transactions on Vehicular Technology*, 68(3):2044–2051, 2019.
- [21] Matrice 100 specs. <https://www.dji.com/matrice100/info>, Mar. 2020.
- [22] Hskt technology specifications. <https://www.indotraq.com/>, Mar. 2020.
- [23] White rabbit switch. <https://sevensols.com/index.php/products/white-rabbit-switch/>.
- [24] E. Hamed, H. Rahul, M. Abdelghany, and D. Katabi. Real-time distributed mimo systems. In *2016 Proceedings of ACM SIGCOMM Conference*, page 412–425, 2106.
- [25] S. Corum, J. D. Bonior, R. C. Qiu, N. Guo, and Z. Hu. Evaluation of phase error in a software-defined radio network testbed. In *2012 Proceedings of IEEE Southeastcon*, pages 1–4, Mar. 2012.
- [26] Neel Pandeya. Synchronization and MIMO capability with USRP devices. [https://kb.ettus.com/Synchronization\\_and\\_MIMO\\_Capability\\_with\\_USRP\\_Devices](https://kb.ettus.com/Synchronization_and_MIMO_Capability_with_USRP_Devices), May 2016.
- [27] W. Khawaja, O. Ozdemir, F. Erden, I. Guvenc, and D. W. Matolak. UWB air-to-ground propagation channel measurements and modeling using UAVs. In *2019 IEEE Aerospace Conference*, pages 1–10, 2019.