sUAS Position Estimation and Fusion in GPS-Degraded and GPS-Denied Environments using an ADS-B Transponder and Local Area Multilateration

Robert Sherman Larson

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science in Aeronautics & Astronautics

University of Washington

2017

Committee:

Christopher Lum

Juris Vagners

Program Authorized to Offer Degree:
Aeronautics & Astronautics

University of Washington

Abstract

sUAS Position Estimation and Fusion in GPS-Degraded and GPS-Denied Environments
using an ADS-B Transponder and Local Area Multilateration

Robert Sherman Larson

Chair of the Supervisory Committee:
Research Assistant Professor Christopher Lum
William E. Boeing Department of Aeronautics and Astronautics

An Unmanned Aerial Vehicle (UAV) and a manned aircraft are tracked using ADS-B transponders and the Local Area Multilateration System (LAMS) in simulated GPS-degraded and GPS-denied environments. Several position estimation and fusion algorithms are developed for use with the Autonomous Flight Systems Laboratory (AFSL) TRansponder based Position Information System (TRAPIS) software. At the lowest level, these estimation and fusion algorithms use raw information from ADS-B and LAMS data streams to provide aircraft position estimates to the ground station user. At the highest level, aircraft position is estimated using a discrete time Kalman filter with real-time covariance updates and fusion involving weighted averaging of ADS-B and LAMS positions. Simulation and flight test results are provided, demonstrating the feasibility of incorporating an ADS-B transponder on a commercially-available UAS and maintaining situational awareness of aircraft positions in GPS-degraded and GPS-denied environments.

The views expressed are those of the author and do not reflect the official policy or position of the US Air Force, Department of Defense or the US Government.

TABLE OF CONTENTS

LIST OF FIGURES

Figure Number Page

LIST OF TABLES

Table Number Page

GLOSSARY

ADS-B: Automatic Dependent Surveillance - Broadcast

AGL: Above Ground Level (aircraft altitude)

ANPC: Advanced Navigation and Positioning Corporation

COA: Certificate of Authorization

COTS: Commercial Off-The-Shelf

E-LSA: Experimental Light Sport Aircraft

FAA: Federal Aviation Administration

FAR: Federal Aviation Regulations

GA: General Aviation; all non-commercial civil aviation

GCS: Ground Control Station

GPS: Global Positioning System

HIL: Hardware in the Loop

ICAO: International Civil Aviation Organization

JCATI: Joint Center for Aerospace Technology Innovation

KDLS: Columbia Gorge Regional/The Dalles Municipal Airport

LAMS: Local Area Multilateration System

MFOC: Mobile Flight Operations Center

MSL: Mean Sea Level (aircraft altitude)

NAS: National Airspace System

SSR: Secondary Surveillance Radar

SSRTM: Space Shuttle Radar Topography Mission

TCAS: Traffic Collision Avoidance System

TRAPIS: TRansponder based Position Information System

(S)UAS: (Small) Unmanned Aerial System

VHF: Very High Frequency (30 MHz to 300 MHz)

VOR: VHF Omnidirectional Range

ACKNOWLEDGMENTS

I would like to thank Dr. Christopher Lum for his mentorship and guidance, and for allowing me the opportunity to work on this project as a member of the Autonomous Flight Systems Laboratory (AFSL). I would also like to thank Professor Emeritus Juris Vagners for serving on my thesis committee, and for developing this project along with Dr. Andy Von Flotow of Hood Technology Corp. This research was made possible through generous funding from the Joint Center for Aerospace Technology Innovation (JCATI). Furthermore, the research would not have been possible without support from our industry partners: Hood Technology Corp., Sagetech Corp., and the Advanced Navigation & Positioning Corp. (ANPC). I would also like to thank additional members of the AFSL who helped make this work possible: Ward Handley, Gage Winde, Henry Qin, Ryan Valach, Emil Caga-anan, Marissa Reid, Anupam Gupta, Zach Caratao, Selina Lui, and ZhenZhen Su. Additionally, I'd like to thank the Aeronautics and Astronautics Department graduate advisors Ed Connery and Leah Panganiban for helping me ensure that all program requirements were met. I am extremely thankful for the guidance of my professors who facilitated my continued learning in pursuit of this challenging but rewarding degree. Finally, I would like to thank my family and friends for their constant support and encouragement throughout this journey.

DEDICATION

For my grandfather Ken Sherman

Thank you for your endless inspiration and encouragement

Chapter 1

INTRODUCTION

In recent years, the small unmanned aerial systems (sUAS) market has expanded rapidly. As sUAS and associated technologies become increasingly popular and more accessible to the general public, the additional risks associated with these vehicles must be considered. Currently, the Federal Aviation Administration (FAA) estimates that sUAS purchases by hobbyists will grow from 1.9 million in 2016 to 4.3 million by 2020, and UAS purchased for commercial uses will grow from 600,000 in 2016 to 2.7 million by 2020 [9]. After coupling these forecasts with the increasing complexity and capabilities of commercially-available sUAS, the increased risks for accidents involving sUAS and other aircraft become clear. Although the FAA has begun attempts to regulate how sUAS are flown and used for commercial purposes through Part 107 [12] certification, sUAS hobbyists occupy a unique sphere within the aviation industry. Many hobbyists now have access to powerful sUAS but lack the airspace and aircraft operational knowledge required of all manned aircraft pilots, private and commercial. This lack of knowledge has most-recently been evidenced in several near-collisions between manned aircraft and sUAS [31, 28]. Improving the situational awareness of sUAS operators and manned aircraft pilots operating in the same airspace as sUAS is therefore imperative if further incidents are to be avoided.

1.1 Problem Statement

1.1.1 sUAS ADS-B Integration

The proliferation of sUAS technologies comes at a time when the FAA is seeking to rapidly change the National Airspace System (NAS), specifically the way in which aircraft interact with one another and Air Traffic Control (ATC). Recently, the FAA has provided a framework by which all manned aircraft will be required to meet basic equipment requirements by 2020 [10]. Known as NextGen, the FAA mandate requires manned aircraft to be equipped with Automatic Dependent Surveillance - Broadcast (ADS-B) transponders by the year 2020. With an ADS-B transponder, aircraft will be able to broadcast their GPS position to ATC as well as other aircraft flying in the vicinity. Overall, the implementation of this transponder network will provide ATC and pilots with increased situational awareness, thereby increasing the safety of normal flight operations within the United States. If these ADS-B transponders can be manufactured in smaller form factors, then it could be possible for sUAS vehicles to make use of ADS-B transponders as well. This research looks to implement a small ADS-B transponder on a commercially-available sUAS for integration into the NAS. The flight testing associated with this integrated ADS-B transponder is the subject of Chapter 7.

1.1.2 GPS-Degraded and Denied Operations

Although ADS-B transponders will allow improvement of situational awareness among manned aircraft pilots and potentially sUAS operators, ADS-B operation is dependent upon availability of GPS signal. sUAS and manned aircraft alike have become largely dependent upon GPS for a variety of tasks, from navigation to instrument approaches. Although GPS signal is usually available across the continental United States, availability of GPS can be easily denied through illegal jamming, military exercises, or variations in the satellite constellation orientation. As a result of the possibility of GPS signal degradation or denial, alternative methods of aircraft tracking must be available for operations conducted in congested airspace. Furthermore, position estimation algorithms must be developed in order to handle cases in which intermittent GPS information is available, and these algorithms must be able to form accurate, consistent estimates of aircraft position during abnormal GPS operation. This research looks to implement several estimation and data fusion algorithms to generate vehicle position estimates from GPS data associated with ADS-B data streams coupled with estimated positions from a ground-based tracking system, the Local Area Multilateration System (LAMS). The flight test results of these estimation and fusion algorithms are the subject of Chapter 6.

1.2 Literature Review

Since ADS-B and its associated technologies represent a relatively nascent field of aeronautics, past research on ADS-B is necessarily limited, and this JCATI-funded study represents the first AFSL work with such technologies. Although ADS-B research is limited, several sources proved helpful in guiding the direction and scope of this research. These sources include research from the AFSL and outside entities.

1.2.1 Previous AFSL Work

In the past, AFSL work has largely focused on development of algorithms required for a variety of sUAS applications. While many of these algorithms have focused on sUAS applications that are outside the scope of this study, others provided information that was relevant to the work presented in this thesis. Specifically, AFSL work related to maintaining sUAS situational awareness near congested and restricted airspace provided information regarding tracking algorithms and error propagation [33]. The algorithm development associated with this collision avoidance research proved essential to designing appropriate tracking algorithms for the research associated with this thesis. Furthermore, the ADS-B transponder payload used for the research presented in this thesis was developed as detailed in [16]. The results and products of the previous AFSL ADS-B payload research and associated ground testing were used directly in this study during flight testing.

1.2.2 Related Work

In addition to the previous work conducted by members of the AFSL, several major studies have been conducted in similar vein as the research presented in this paper. In 2009, researchers at the University of North Dakota presented software-in-the-loop simulations for an sUAS sense and avoid algorithm which made use of ADS-B information [24]. Another 2009 study involved an ADS-B based collision avoidance system to be used by sUAS in airspace with other unmanned and manned aircraft operating simultaneously [18]. A 2013 study investigated the possibility of incorporating ADS-B transponders on sUAS and presented a case study to include recommendations for ADS-B regulations regarding sUAS aircraft [32]. More recently, researchers investigated additional sense and avoid algorithms with access to multiple data streams to include traffic collision avoidance system (TCAS) and ADS-B information [27]. While studies such as these have largely focused on future regulations and algorithms for operating sUAS in airspace shared with other sUAS and manned aircraft, the research presented in this paper was focused on demonstrating the use of an ADS-B transponder on a commercially-available sUAS and tracking the aircraft in real time with ADS-B and secondary LAMS unit for GPS-degraded and GPS-denied operations.

1.3 Scope of Work

The work presented in this thesis represents a portion of the total work conducted by the AFSL in fulfillment of a JCATI grant established to research safe integration of sUAS into the NAS, specifically in GPS-degraded or GPS-denied environments. The groundwork for this thesis was presented in initial research conducted by former AFSL graduate student Ward Handley, to include design and production of an ADS-B transponder payload and associated ground testing as detailed in [16]. The research presented in this thesis focused on integration of this transponder payload onto a sUAS as well as software-based estimation and fusion of aircraft position information provided by ADS-B and LAMS data streams.

Chapter 2
BACKGROUND

2.1 National Airspace System

All air traffic in the United States, manned and unmanned, flies within the National Airspace System. The NAS was originally designed in order to ensure that flights could be completed safely, expeditiously, and efficiently at a time when unmanned aircraft operations were a distant, unfathomable possibility [1]. Historically, commercial aviation has accounted for the largest percentage of NAS usage among manned aircraft, and as a result the FAA uses commercial air travel as a benchmark to project industry growth and forecast airspace usage. By the latest reports, the FAA expects the international commercial aviation market to grow at 2.6% per year and the domestic market to grow by more than 50% over the next two decades [9]. After coupling these estimates with projected growth in the burgeoning UAS industry, it becomes evident that the NAS will see rapid changes in traffic type and volume over the next several decades. These changes come at a time when the NAS is dependent upon radar and VHF omnidirectional range (VOR) technologies that have seen few major developments since their inception in the 1950s [1]. In order to ensure that manned and unmanned aircraft can operate simultaneously in a safe manner, the NAS will require significant changes.

2.1.1 Airspace Environments

In order to understand the limitations imposed by the current structure and management of the NAS, a basic understanding of the NAS is necessary. The NAS encompasses all of the airspace above the United States, in addition to all of the airports and navigational facilities required for the safe operation of aircraft. It was originally conceived and designed in the

1970s as a response to increased air traffic and federal mandates for the FAA [1]. The system relies on a network of navigational facilities, primarily VOR stations and surveillance radars, the newest of which were implemented in the late 1980s [13]. Although the infrastructure is dated and rapidly becoming obsolete, the NAS continues to provide a structured environment in which aircraft can operate safely.

Of primary concern to pilots and UAS operators alike is the classification of airspace under the NAS. Airspace is classified based on the volume and type of air traffic experienced in a given region. For example, the airspace in the immediate vicinity of the greater Seattle metropolitan area is much more heavily controlled and monitored than many airspaces in eastern Washington and less-populated areas. In order for pilots to operate legally and safely within the NAS, they must understand and follow the rules and regulations established in Federal Aviation Regulations (FAR). These rules provide procedures and guidelines for how different types of aircraft must be operated within the NAS, specifically when flown in different types of airspaces. Furthermore, these regulations provide requirements for licenses and certifications that pilots must hold to operate within certain airspaces. An example diagram of the different types of airspaces defined within the NAS is shown in Figure 2.1.

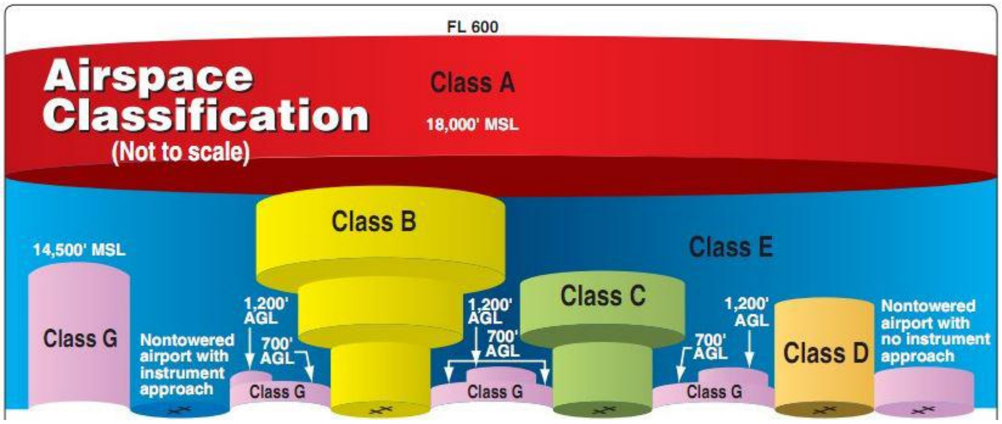


Figure 2.1: NAS Airspace Diagram [4].

Aside from the procedures that pilots must follow when operating aircraft in different types of airspace, there are equipment requirements that must be met as well. For instance, commercial aircraft which operate at altitudes above 18,000 feet above Mean Sea Level (MSL) in Class A airspace are required to have additional equipment not required for most general aviation (GA) aircraft operating at altitudes well below the boundaries of Class A airspace [11]. Similarly, aircraft operating in congested Class B airspace near major airports are required to have specific equipment not required of other aircraft [14].

2.1.2 ATC Infrastructure and Limitations

In order to ensure that aircraft operations are conducted safely within the NAS, Air Traffic Control (ATC) facilities are maintained throughout the United States. ATC facilities range from control towers at regional airports to approach control stations at large airports, to regional traffic control, with personnel responsible for tracking aircraft across the United States. At its core, the purpose of ATC is to prevent collisions between aircraft and to ensure that aircraft operate properly and efficiently within the NAS [11]. Although most pilots currently use Global Positioning System (GPS) receivers as a primary means of navigation, ATC facilities continue to rely on an outdated infrastructure replete with coverage gaps and known potential for external interference. Like the NAS, the ATC infrastructure is limited in many ways and in need of significant updates.

The various equipment and pilot certifications required by the FARs were originally designed for manned aircraft operations, without the expectation that the UAS industry would grow as quickly as it has. Similarly, the ATC system was designed with manned aircraft operations as the focus, with pilot education and accountability serving as the primary guarantors of flight safety within the NAS. However, UAS industry development has rapidly outpaced the development of UAS regulations. As a result, the availability of UAS to consumers has led to a potentially dangerous situation in which UAS operators lacking knowledge of the requirements and regulations associated with the NAS could jeopardize the continued safety of other aircraft operations.

2.2 ADS-B

In order to facilitate modernization of the NAS and ATC infrastructure, new technologies leverage the availability and accessibility of GPS. One such technology has proven particularly useful in monitoring air traffic without the need for traditional radar stations, Automatic Dependent Surveillance Broadcast (ADS-B). ADS-B is a technology which uses the position information from a GPS receiver installed on an aircraft to provide real-time position updates to ATC as well as other aircraft operating in the vicinity with appropriate equipment installed [8].

Current ADS-B technologies combine the broadcasting feature with standard Mode S aviation transponders currently required for aircraft operating in most airspace. The equipment used to broadcast the ADS-B information is known as ADS-B Out, while the equipment used to receive ADS-B information is known as ADS-B In. For the GA market, the cost of an ADS-B Out transponder with associated equipment and installation currently ranges from \$5000 to \$10,000 depending on the model and increased GPS requirements [19].

2.2.1 NextGen Framework

Within the last two years, the FAA has generated requirements for the integration of ADS-B technologies on aircraft operating within the NAS. Known as NextGen, the FAA framework requires that all manned aircraft operating in Class A, B, C, and E (above 10,000 feet MSL) airspaces must be equipped with ADS-B equipment by the year 2020 [10]. The FAA estimates that NextGen implementation will result in reduction of commercial aviation delays of 38% by 2020, a reduction which would amount to estimated savings of \$24 billion and 1.4 billion gallons of fuel [30]. Integration of ADS-B Out technologies will facilitate these savings by improving efficiency of arrivals and departures at airports around the United States, and by providing ATC controllers with more options for routing air traffic enroute to their destinations.

2.2.2 ADS-B Limitations

Although incorporation of ADS-B Out transponders will undoubtedly streamline ATC procedures and allow aircraft to operate safely with greater situational awareness within the NAS, the technology does not come without limitations. Since ADS-B operation is inherently dependent on GPS position information, it is limited to an extent by the availability and integrity of GPS. In recent years, GPS outages over significant portions of the United States have been experienced as a result of military training [26]. Although much less common and highly illegal, GPS jamming to targeted areas can be accomplished through the use of low-cost, low-power units that can be made with readily-available materials, an example of which is shown in Figure 2.2. Aside from planned and unplanned GPS outages, GPS signal integrity is dependent upon the orientation of the GPS satellite constellation, and position accuracy is subject to degradation as a result of location, terrain, weather, and other factors. These limitations of ADS-B associated with GPS availability prompted the research into alternative tracking technologies and development of estimation and fusion algorithms detailed in this thesis.

Figure 2.2: Illegal GPS jamming device [6].

Chapter 3

EQUIPMENT

3.1 ADS-B Payload

The ADS-B payload included the Sagetech XPS-TR transponder unit [29] shown in Figure 3.1 along with several additional components. The ADS-B payload was developed by Ward Handley as detailed in [16]. Interested readers are encouraged to consult the referenced material for detailed information regarding the design and fabrication of the ADS-B payload. The wiring diagram for the ADS-B payload is shown in Figure 3.2, and the major payload components are listed in Table 3.1.

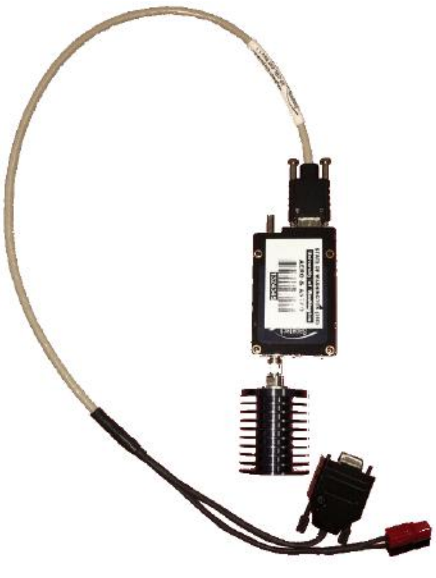


Figure 3.1: XPS-TR Transponder with RF terminator attached [16].

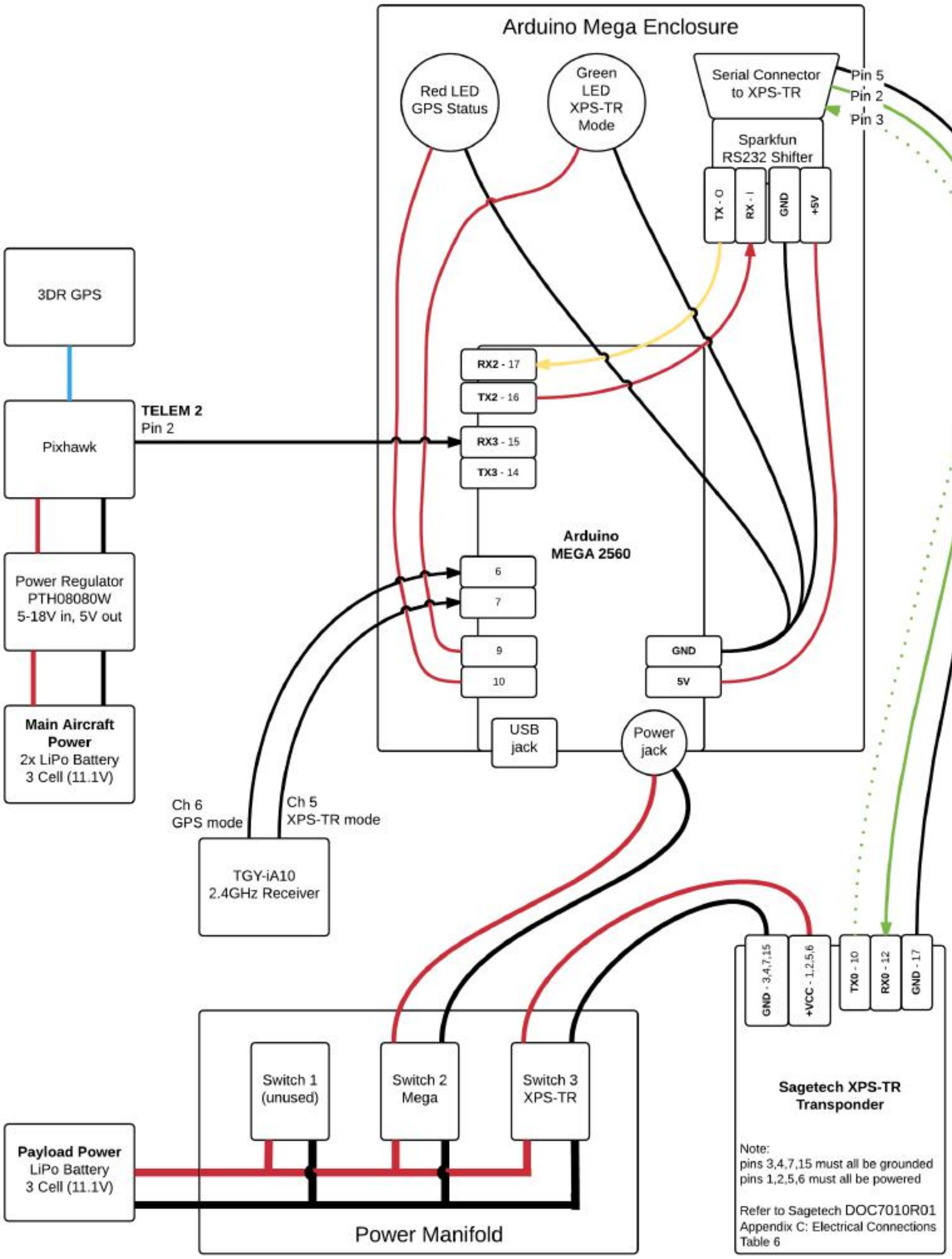


Figure 3.2: TRAPIS avionics wiring diagram [16].

Component	Specification
Transponder	Sagetech XPS-TR [29]
GPS	3DR uBlox [3]
Receiver	Turnigy TGY-iA10 2.4 GHz receiver
Battery	5000 mAh 3S 25C LiPo
Serial Connector	SparkFun RS232 Shifter [2]
Microcontroller	Arduino MEGA 2560 [5]

Table 3.1: ADS-B Payload Components.

3.2 sUAS Components

3.2.1 Aircraft

The sUAS aircraft used for all testing was a Skywalker 1900, nicknamed Leia, with registration number N632UW. The aircraft is shown in Figure 3.3 as-flown during the culminating flight demonstration associated with this research. The aircraft specifications and components are listed in Table 3.2 and Table 3.3.

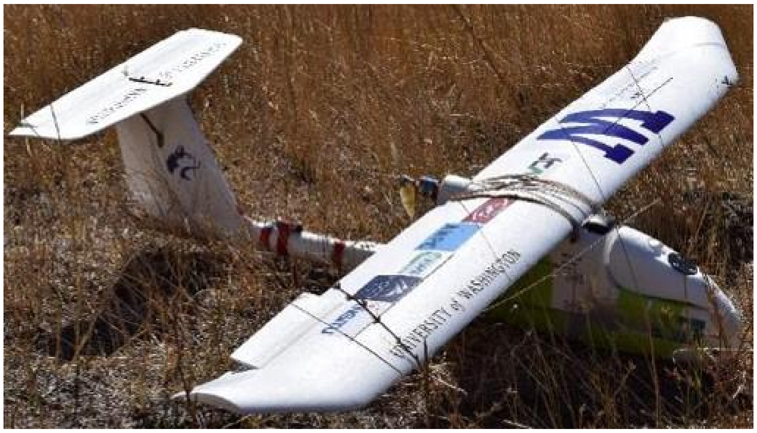


Figure 3.3: Leia Skywalker 1900 sUAS.

Component	Specification
Wingspan	6.23 feet
Length	3.87 feet
Wing Area	4.68 ft^2
Total Weight	6.42 lbs.
Wing Loading	1.37 $\frac{lbs.}{ft^2}$
Endurance	15 minutes
Center of Gravity	17.75 in. aft of nose
Thrust	2.7 lbs.

Table 3.2: Leia Aerodynamic Specifications.

Component	Specification
Battery	5000 mAh 3S 25C LiPo
Motor	Turnigy 1000 KV brushless
Flight Controller	Pixhawk [20]
Firmware	Arduplane [17]
GPS	3DR uBlox [3]
Payload	Transponder payload [16]

Table 3.3: Leia Component Specifications.

3.2.2 GCS

The ground control station used for all guidance, autonomous flight, and telemetry was a dedicated AFSL desktop computer running Windows 8.1 Enterprise. Mission Planner version 3.6.0 was used for all flight testing, and represents the standard ground control station software package used for all AFSL flight operations. Additional specifications of the GCS computer are shown in Table 3.4

3.3 Manned Aircraft

In order to facilitate sUAS transponder testing in a realistic GA airport environment, two manned aircraft were used during the flight demonstration. These aircraft operated in flight testing locations southeast and southwest of the KDLS airfield to ensure geographic

Component	**Specification**
Operating System	Windows 8.1 Enterprise
Processor	Intel Core i7-4790K
RAM	16 GB
GCS Software	Mission Planner v3.6.0

Table 3.4: AFSL GCS Computer Specifications.

separation from all sUAS operations as detailed in Chapter 4. Manned aircraft support was provided by TacAero Inc., an aircraft training center located at the Ken Jernstedt Airfield (4S2) in Hood River, OR.

The two manned aircraft flown for the demonstration were a Vans RV-12, N484TA, and a Cessna C172SP, N562AE. These two aircraft were chosen in order to provide a comparison between disparate aircraft equipped with different types of transponders. The Vans RV-12 is a kit aircraft that is categorized as an Experimental Light Sport Aircraft (E-LSA), a category for which the FAA has designated specific weight and performance limitations. The RV-12 owned by TacAero Inc. was factory-built at Synergy Air in Eugene, OR and utilized an ADS-B-compliant transponder. The Vans RV-12 used during the flight test is shown in Figure 3.4(a).

The Cessna C172SP is a high-wing, single engine general aviation aircraft that has remained a popular training aircraft since its inception in 1955. Unlike the Vans RV-12, the Cessna 172SP owned by TacAero Inc. does not use an ADS-B-compliant transponder, and instead operates a standard Mode C transponder with altitude encoding capability. The Cessna 172 used during the flight test is shown in Figure 3.4(b).

Within the scope of the FAA NextGen 2020 ADS-B transponder requirements, these two aircraft provided a direct comparison of aircraft tracking capabilities using ADS-B and non-ADS-B methods. Furthermore, the presence of manned aircraft during sUAS testing was intended to resemble scenarios in which sUAS and manned aircraft will operate simultaneously in nearby airspace. Based on the rapid and continued growth of the sUAS industry,

(a) Vans RV-12.

(b) Cessna C172SP.

Figure 3.4: RV-12 and Cessna 172 used for the flight demonstration.

such scenarios will occur with increasing frequency in the near future.

3.3.1 Secondary GPS Units

In order to validate the data gathered from the manned aircraft on-board GPS units, the AFSL provided two secondary GPS units that were flown on the manned aircraft. These secondary units used the same GPS hardware used on the sUAS, to include the 3DR uBlox GPS unit and a Pixhawk flight controller. Data was gathered using these hardware in the loop (HiL) GPS units over the course of the full flights for the Cessna 172 and Vans RV-12. An example of the secondary GPS units is shown in Figure 3.5

3.4 LAMS Description

The Local Area Multilateration System (LAMS), is a system developed by the Advanced Navigation and Positioning Corporation (ANPC) as shown in Figure 3.6. The system uses several antennas to provide aircraft position and velocity information derived from triangulation and Doppler shift information. The altitude associated with a specific target aircraft is provided by the Mode C or Mode S transponder installed on that aircraft, and this information is received at the LAMS ground station and presented to the system user.

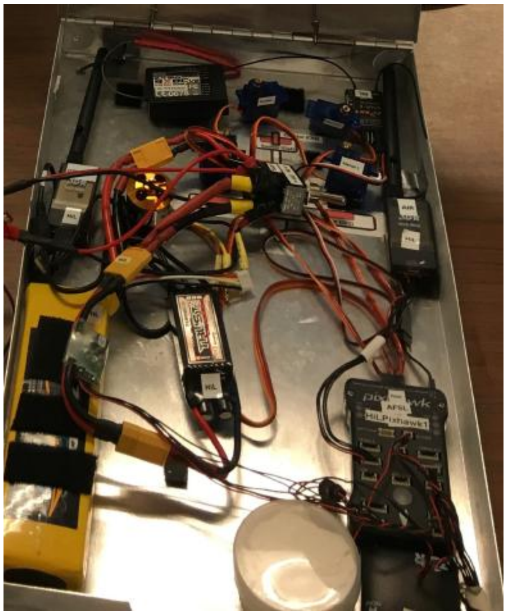


Figure 3.5: Secondary GPS HiL unit.

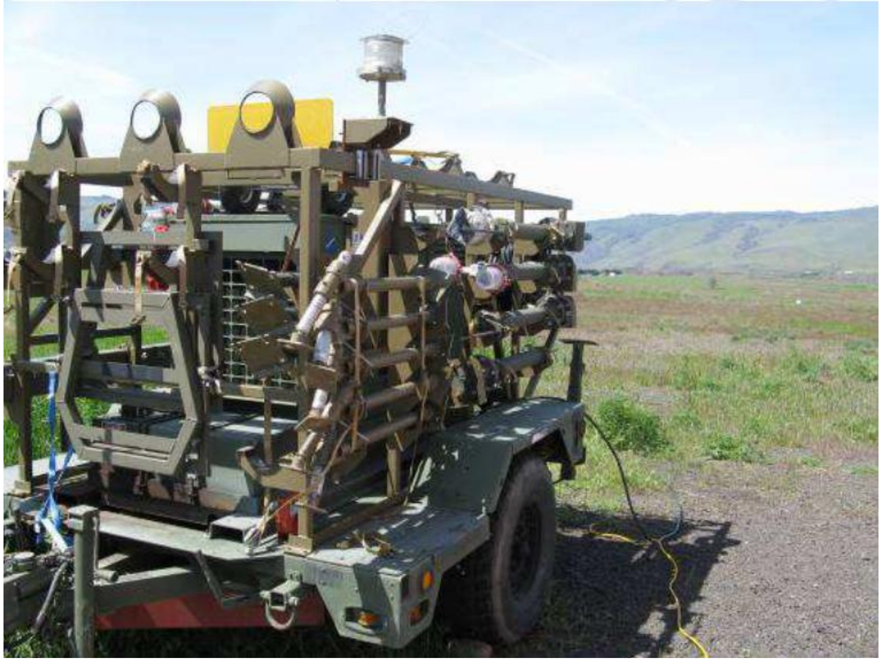


Figure 3.6: Local Area Multilateration System (image courtesy of ANPC).

When compared to traditional methods of local aircraft tracking, including technologies such as secondary surveillance radar (SSR), the LAMS requires much less permanent infrastructure. Furthermore, the LAMS can be set up in several hours by a dedicated ground team and in that sense is a rapidly-deployable technology which provides sufficient localized aircraft tracking capabilities.

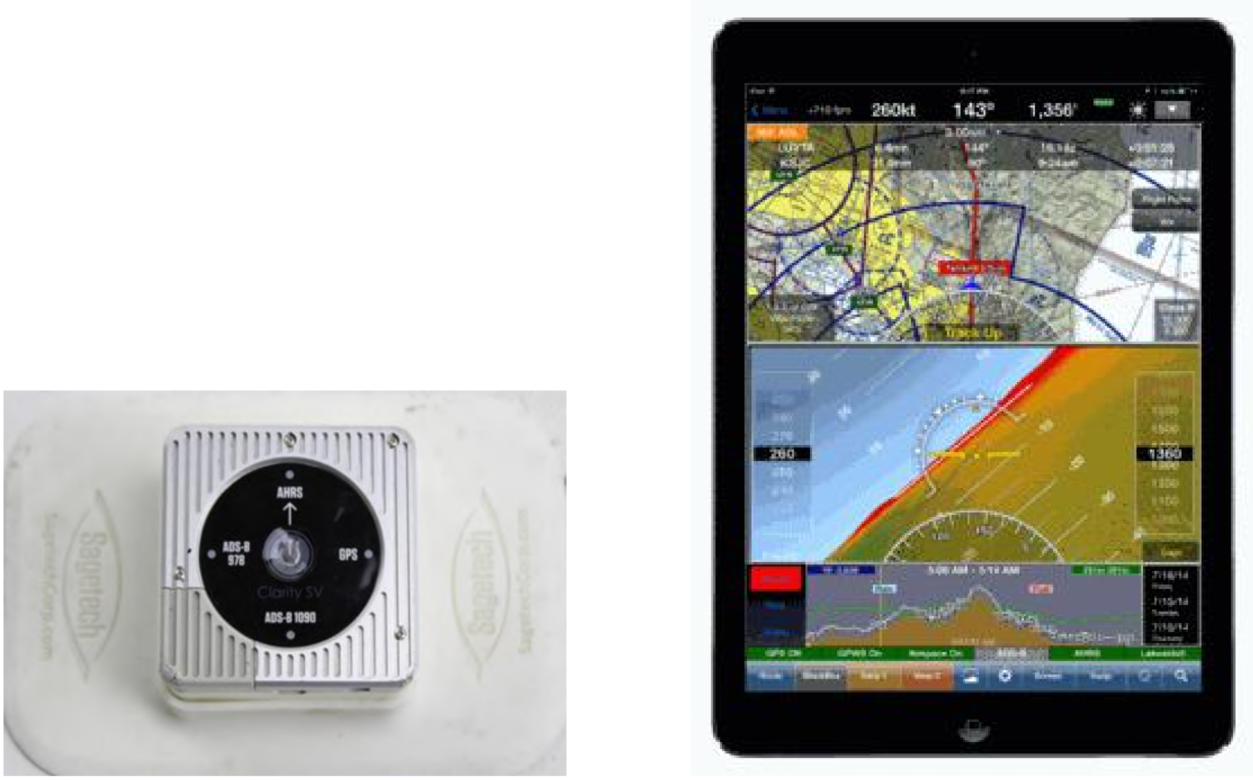
3.5 Tracking Equipment

3.5.1 ADS-B In Hardware

In order to receive ADS-B information associated with the aircraft used during the flight testing, ADS-B In hardware and associated software was required. The Sagetech Clarity ADS-B In receiver was used to gather the required ADS-B information. Initially, this information was displayed on an AFSL iPad running the WingX Pro7 application. WingX Pro7 allowed for real-time tracking of ADS-B equipped aircraft on a moving-map aviation sectional display. The Clarity receiver is shown in Figure 3.7(a) with a corresponding screen capture of the WingX Pro7 application shown in Figure 3.7(b).

3.5.2 TRAPIS

The TRansponder based Position Information System (TRAPIS) is a software package developed by members of the AFSL which was used for simultaneous ADS-B and LAMS aircraft tracking. The TRAPIS software was developed in C# using Microsoft Visual Studio, and all associated code was thoroughly tested. The TRAPIS software package allows a ground station user to view air traffic in real-time on a moving map display and pair aircraft ADS-B and LAMS data streams using an intuitive Graphical User Interface (GUI) as shown in Figure 3.8. Once the TRAPIS user pairs the ADS-B and LAMS data streams for a given aircraft, the estimation and fusion algorithms detailed in Chapter 5 begin generating estimates of the aircraft position. The fused estimates for an aircraft are also displayed on the TRAPIS moving map, allowing the user to track aircraft position estimates in real-time.



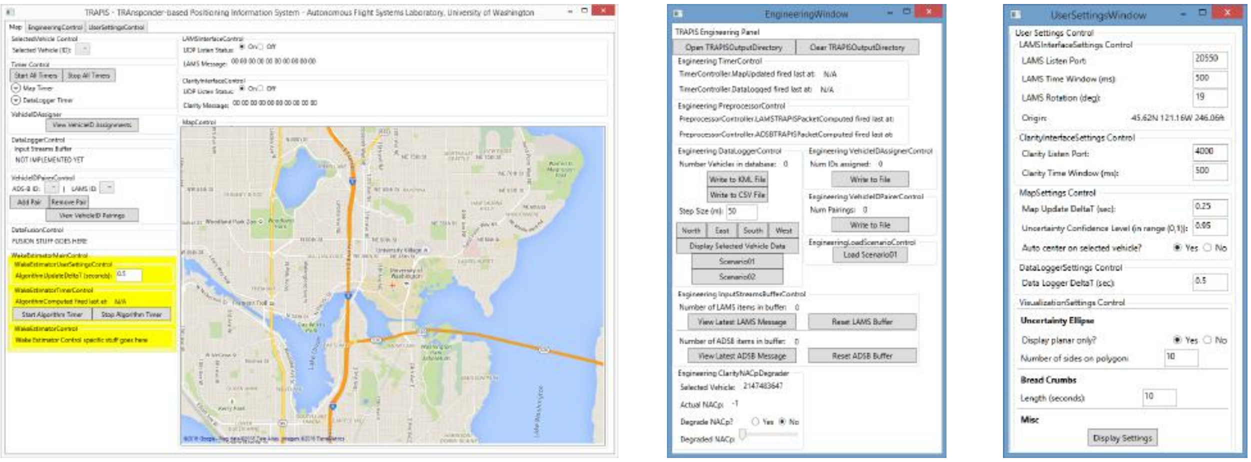
(a) Sagetech Clarity ADS-B receiver.

(b) WingX Pro7 iPad screen capture.

Figure 3.7: Sagetech Clarity ADS-B receiver and WingX Pro7 iPad screen capture.

TRAPIS received ADS-B data through a connection to the Sagetech Clarity ADS-B receiver detailed in the previous section. LAMS data was received through a direct connection to the LAMS station at the KDLS airfield. Marker icons associated with ADS-B and LAMS data streams were assigned different colors in order to make identification and pairing of aircraft position information intuitive for the user. Additionally, ADS-B and LAMS position information gathered through TRAPIS included displays of standard deviation values associated with each data stream

In addition to ADS-B and LAMS tracking capabilities, the TRAPIS software framework allows for additional development of third-party applications. Once such application, the Wake Turbulence Estimator, is detailed in [16]. Further development of TRAPIS will likely



Selected Vehicle (ID):

Input Streams Buffer
NOT IMPLEMENTED YET

ADS-B ID: | LAMS ID:

FUSION STUFF GOES HERE

Algorithm UpdateDeltaT (seconds): 0.5

AlgorithmComputed fired last at: N/A

Wake Estimator Control specific stuff goes here

UDP Listen Status: ● On ○ Off
LAMS Message: 00 00 00 00 00 00 00 00 00 00

UDP Listen Status: ● On ○ Off
Clarity Message: 00 00 00 00 00 00 00 00 00 00

TimerController.MapUpdated fired last at: N/A
TimerController.DataLogged fired last at: N/A

PreprocessorController.LAMSTRAPISPacketComputed fired last at:
PreprocessorController.ADSBTRAPISPacketComputed fired last at:

Number Vehicles in database: 0
Step Size (m): 50

Num IDs assigned: 0

Num Pairings: 0

Number of LAMS items in buffer: 0
Number of ADSB items in buffer: 0

Selected Vehicle: 2147483647
Actual NACp: -1
Degrade NACp? ○ Yes ● No
Degraded NACp:

LAMS Listen Port: 20550
LAMS Time Window (ms): 500
LAMS Rotation (deg): 19
Origin: 45.62N 121.16W 246.06ft

Clarity Listen Port: 4000
Clarity Time Window (ms): 500

Map Update DeltaT (sec): 0.25
Uncertainty Confidence Level (in range (0,1)): 0.95
Auto center on selected vehicle? ● Yes ○ No

Data Logger DeltaT (sec): 0.5

Display planar only? ● Yes ○ No
Number of sides on polygon: 10

Length (seconds): 10

Figure 3.8: TRAPIS GUI and controls [16].

include additional utilities for safe integration of sUAS operations within the NAS. The modular nature of the TRAPIS software will allow for such additions to be made without the need for significant overhauls of current system functionality.

Chapter 4

EXPERIMENTAL METHODS

4.1 Experimental Setup

4.1.1 Initial Flight Testing

Before the flight demonstration at KDLS was completed, several local flight tests were conducted with the test sUAS carrying the TRAPIS payload. These initial flight tests were carried out in order to ensure that all required tests could be completed successfully while tracking the sUAS with an ADS-B In receiver. Additionally, these tests served to validate TRAPIS software performance in a real-world environment with manned aircraft operating at various altitudes near the test airspace. The test flights were completed during three separate flight testing excursions on August 25th, September 9th, and September 16th, 2016.

All preliminary flight testing was conducted at Meadowbrook Farms, a test airspace currently used for all AFSL local sUAS flight testing located in North Bend, WA. The test site is a 64 acre field located 25 miles east of downtown Seattle at 47.518868 N, 121.802444 W and shown in Figure 4.1. The flight testing location can be reached from the University of Washington in 40 minutes by car.

The site was selected for additional reasons outside of proximity and travel time, and the location proved valuable for both sUAS flight testing and transponder testing considerations. Prior to the implementation of 14 CFR Part 107 on August 29, 2016, all AFSL flight operations were conducted under a Certificate of Authorization (COA) approved by the FAA. The terms of the COA required that all AFSL flight operations take place in Class G airspace below 400 feet AGL with aircraft weighing less than 55 lbs. Once these COA restrictions were accounted for, further precautions were taken to ensure that all transponder testing would not interfere with commercial and private aircraft operations. Since no previous test

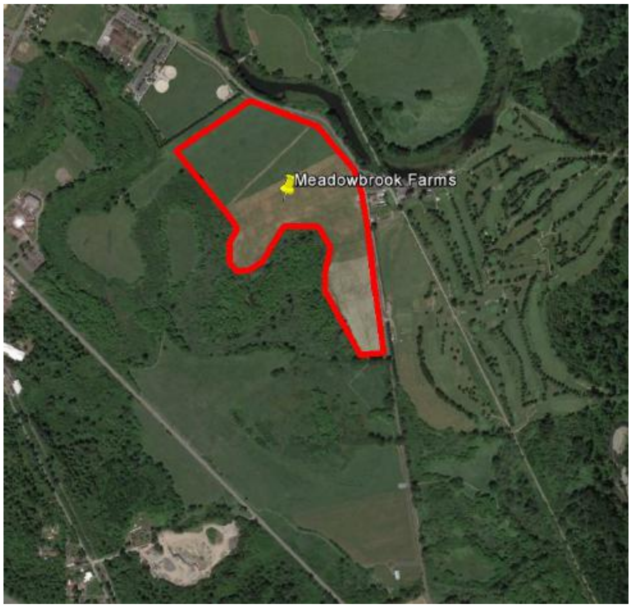


Figure 4.1: The Meadowbrook Farm flight testing location in North Bend, WA.

data had been gathered to ensure the validity of the transponder altitude reporting capabilities, the decision was made to conduct all transponder testing outside of the confines of the Seattle Class B airspace. Meadowbrook Farms is located outside of the Class B airspace and allowed test crews to fly sUAS aircraft up to 400 feet AGL while remaining within Class G airspace.

The first of the initial flight tests was accomplished on August 25th, 2016. The goal of the first test was to verify that the TRAPIS payload would function properly while installed on the sUAS airframe. In order to accomplish this goal, two ground tests were conducted wherein the TRAPIS payload was turned on while installed inside the aircraft payload bay. During these tests it was shown that the ADS-B Out capability of the transponder was functioning properly, and the aircraft was seen using the ADS-B In receiver coupled with the iPad WingX Pro application. After the two ground tests were completed, flight tests were conducted in order to show that the sUAS aircraft could follow a pre-designated flight path while the transponder was operating as detailed in Chapter 6. In order to test the autonomous flight path tracking abilities of the test aircraft, a rectangular flight path was established as shown in Figure 4.2. Each leg of the rectangular flight path was 0.15 NM in length.

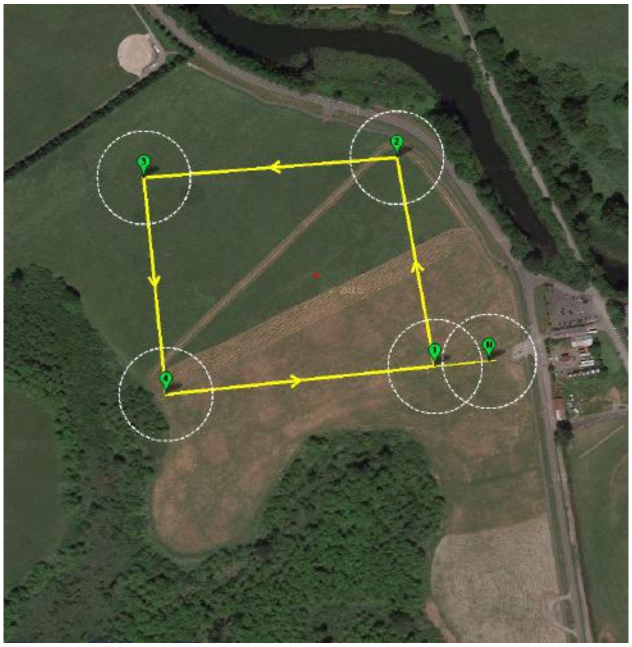


Figure 4.2: Initial test flight path.

4.1.2 KDLS Test Site Selection

In order to ensure that all operations were conducted in accordance with COA and FAA requirements, and in the interest safety for persons and equipment involved in the flight test, exhaustive site studies were conducted. The site studies led to the development of a detailed test plan which required coordination with multiple agencies and personnel.

After ground testing was completed in the spring of 2016, preparations began for the final TRAPIS flight demonstration. Based on the results of the ground tests coupled with ADS-B and LAMS limitations, it was known that line-of-sight would be required between the sUAS and the KDLS ground station for all flight operations. Additionally, COA requirements stipulated that all flight operations would need to be conducted at altitudes less than 400 feet AGL. Furthermore, the COA required that all sUAS operations in Class G airspace near a non-towered airport with published instrument flight procedures must occur more than 3 nautical miles from the airport reference point. In order to determine suitable flight test locations subject to these criteria, an RF site survey was conducted. This survey was completed using Radio Mobile RF analysis freeware coupled with guidance provided by ANPC personnel. The software allowed coverage maps to be created for the areas surrounding the KDLS airport subject to RF specifications of the LAMS unit and the XPS-TR transponder unit.

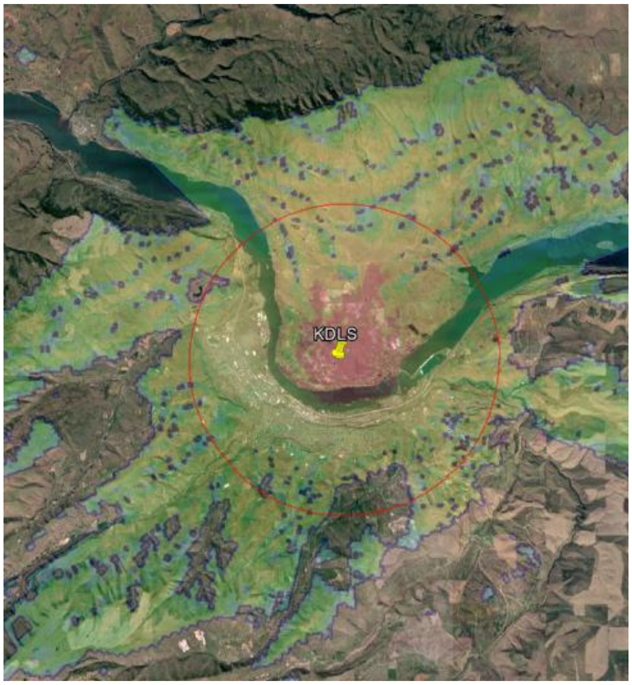


Figure 4.3: 200 feet AGL RF coverage map.

After it was determined that flight testing would be conducted at altitudes ranging from 200 feet AGL to 400 feet AGL at least 3 NM from the airport to comply with COA restrictions, corresponding RF coverage plots were created. These coverage plots ensured that line-of-sight and adequate RF signal strength would be maintained between the sUAS transponder and the LAMS station at the distances and altitudes required for the flight test. Ultimately, potential flight test locations were determined by coverage maps generated for the 200 feet AGL condition, since this altitude proved to be the most restrictive when considering obstacles and ground clutter. RF signal strength estimates at possible test sites were determined by using a simulated stationary antenna located at the KDLS airfield to represent the LAMS station. The power and frequency specifications of the real LAMS station were modeled in order for this simulated antenna to accurately model the LAMS station. An additional antenna with power and frequency specifications matching those of the XPS-TR transponder was modeled. Terrain information was gathered from the data archives of the Space Shuttle Radar Topography Mission (SSRTM) for the areas surrounding the KDLS airfield. Once this terrain data was gathered it was provided to the Radio Mobile software, and the transponder antenna elevation was set at 200 feet AGL. Based on these inputs, RF coverage maps were generated for the area surrounding KDLS.

An example coverage map for the 200 foot AGL flight case is shown in Figure 4.3. The yellow pin indicates the location of the KDLS airport reference point, and the red circle shows the 3 NM radius around the airport. The coverage map shows LAMS signal strength as seen by the transponder antenna at 200 feet AGL, where red indicates the strongest signals at -43 dBm, and blue indicates the weakest signals at -83 dBm. Areas for which there is no color overlay indicate regions in which the transponder antenna would have no line-of-sight with the LAMS station at 200 feet AGL. Since the KDLS airport is situated in the Columbia River Gorge, ridges to the north, south, and west of the airport prevented the transponder antenna from having line-of-sight with the LAMS station at 200 feet AGL at distances greater than approximately 5 NM from the airport. Coverage maps generated at higher altitudes showed increased area coverage with stronger signals up to and including the maximum allowed altitude of 400 feet AGL. Since site selections were made based on the most restrictive altitude conditions, options for viable flight test locations were confined to those sites between 3 and 5 NM from the airport for which coverage maps predicted adequate signal strength at 200 feet AGL.

Once the RF coverage plots were completed, additional studies were completed to define potential test sites. These studies focused on the accessibility of test locations by vehicle, estimated travel time between the KDLS airport and the proposed test sites, and suitability of terrain for flight test operations. While some general information was gathered regarding test sites and surrounding terrain during the spring 2016 ground test of the TRAPIS system, detailed site surveys were conducted by analyzing terrain data and available ground-level views in Google Earth. Initially, twelve candidate test locations were selected at distances between 3 and 5 NM from the airport. These candidate locations can be seen in Figure 4.4. The two red circles in the figure reflect 3 and 5 NM radii from the KDLS airfield, and each of the yellow pins indicates the location of a candidate flight test location. Each of the black polygons associated with the figure indicate the boundaries of a candidate test airspace.

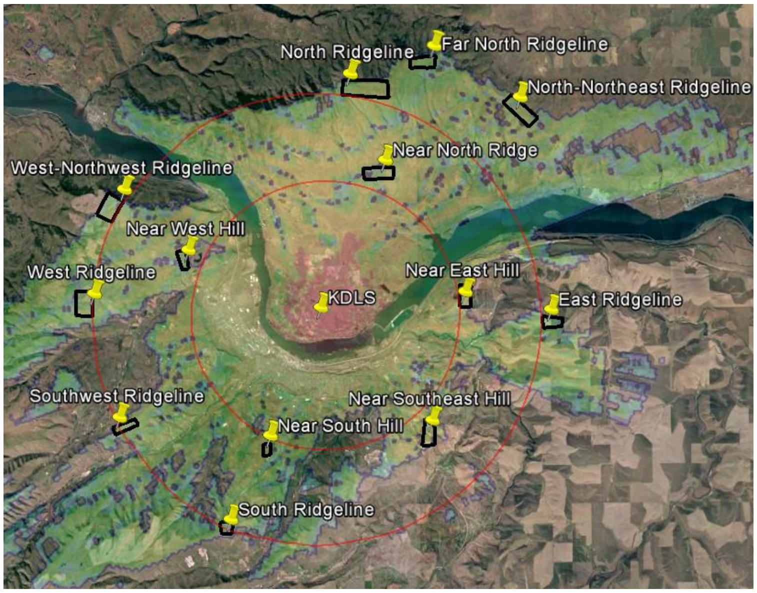


Figure 4.4: Initial flight test candidate locations.

After the initial site selections were completed, further analysis was conducted to find the top three potential test sites for further consideration. The three sites selected for final consideration were two sites to the northeast of the KDLS airfield, and one test site due east of the KDLS airfield as shown in Figure 4.5. Each of the three final candidate locations ensured line-of-sight could be maintained between the sUAS and the LAMS station at the KDLS airport, and each location contained suitable terrain required to conduct sUAS flight operations.

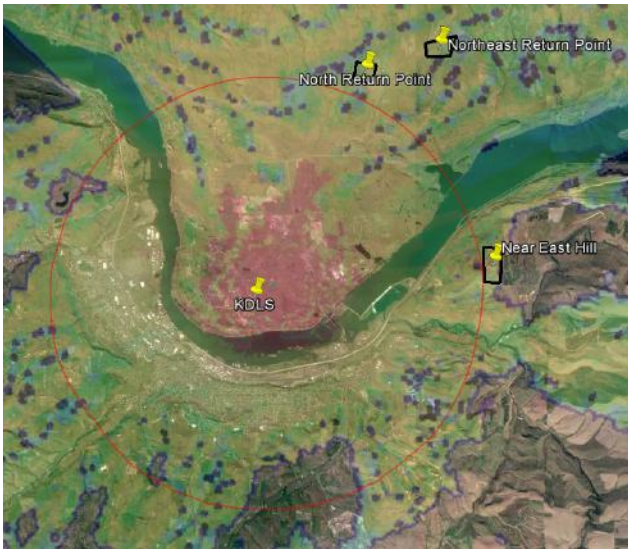


Figure 4.5: Final flight test candidate locations.

Once each of the three final candidate locations was researched further, and once the accessibility of each test site was determined, the northeast test airspace was selected. This test airspace is shown in Figure 4.6, and use of the land for flight testing was granted by the Washington State Parks Department through a Right of Entry Permit. The test site was easily accessed through the use of Dalles Mountain Road, and was located approximately 20 minutes from the KDLS airport by car. The test location provided approximately 155 acres from which to carry out the sUAS operations and met all COA requirements necessary to conduct full testing of the TRAPIS system.

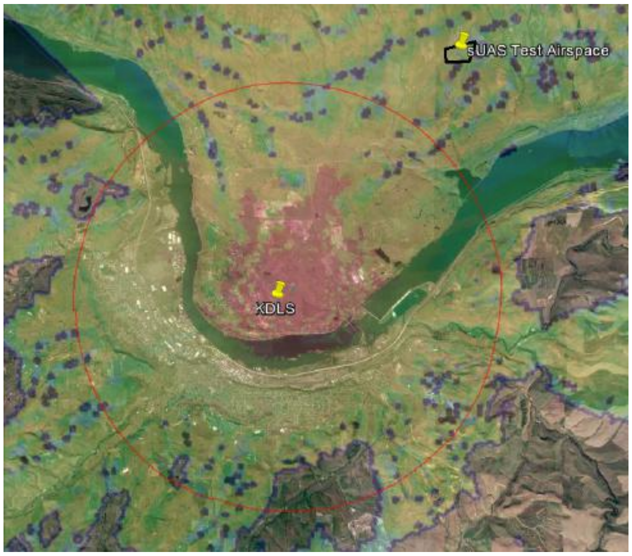
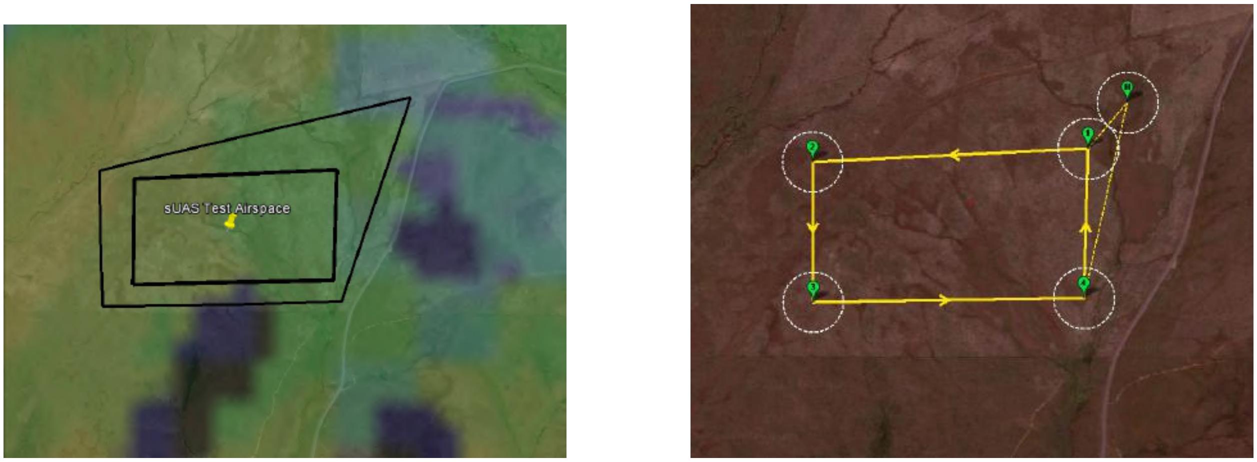


Figure 4.6: Final sUAS flight test airspace.

4.1.3 Test Planning

After the flight test location had been selected, the sUAS flight path was developed. Based on COA requirements and previous flight testing, a rectangular flight path was defined within the test airspace as shown in Figure 4.7(a). This rectangular pattern was designed to match the rectangular flight path used for the preliminary testing of the sUAS at Meadowbrook Farms. The flight path fit within the confines of the approved flight testing airspace with sufficient margins on all sides to prevent airspace breach in the event of diversion from the planned flight path. The long edges of the rectangular flight path were defined to be 0.30 NM, and the short edges were defined to be 0.15 NM for a total flight path distance of 0.90 NM and a total encompassed area of 0.045 square nautical miles.

The sUAS flight path is shown in greater detail in Figure 4.7(b). This image shows the flight path as defined using the Mission Planner software used on the GCS computer for autonomous operations. In consideration of the altitude restrictions imposed under the AFSL COA, the aircraft altitudes were set at 350 feet AGL between the first and second waypoints and 250 feet AGL between the third and fourth waypoints. The flight path was designed for the aircraft to climb between the fourth and first waypoint and descend between the second and third waypoints. Defining the flight path in this manner ensured that the initial flight testing at Meadowbrook Farms closely mirrored the KDLS flight testing.



(a) KDLS sUAS flight test airspace.

(b) Enlarged sUAS flight test path.

Figure 4.7: KDLS sUAS flight test path.

Once the sUAS flight path was defined, the manned aircraft flight paths needed to be defined. At this stage of planning, flight safety considerations had to be taken into account to ensure that sUAS and manned aircraft operations could occur simultaneously while minimizing potential conflicts. In order to ensure adequate separation was maintained between manned and remote aircraft operations, the manned aircraft flight paths were defined to the west and to the southeast of the KDLS airfield, several miles from the sUAS test airspace as shown in Figure 4.8. Each manned aircraft flight path was defined in the same rectangular fashion as the sUAS flight path, with long side lengths of 4.5 NM and short side lengths of 2.7 NM for a total flight path distance of 14.5 NM and a total encompassed area of 12.15 square nautical miles.

In order to ensure that proper vertical separation was maintained between all aircraft participating in the flight demonstration, altitudes were established for the manned aircraft

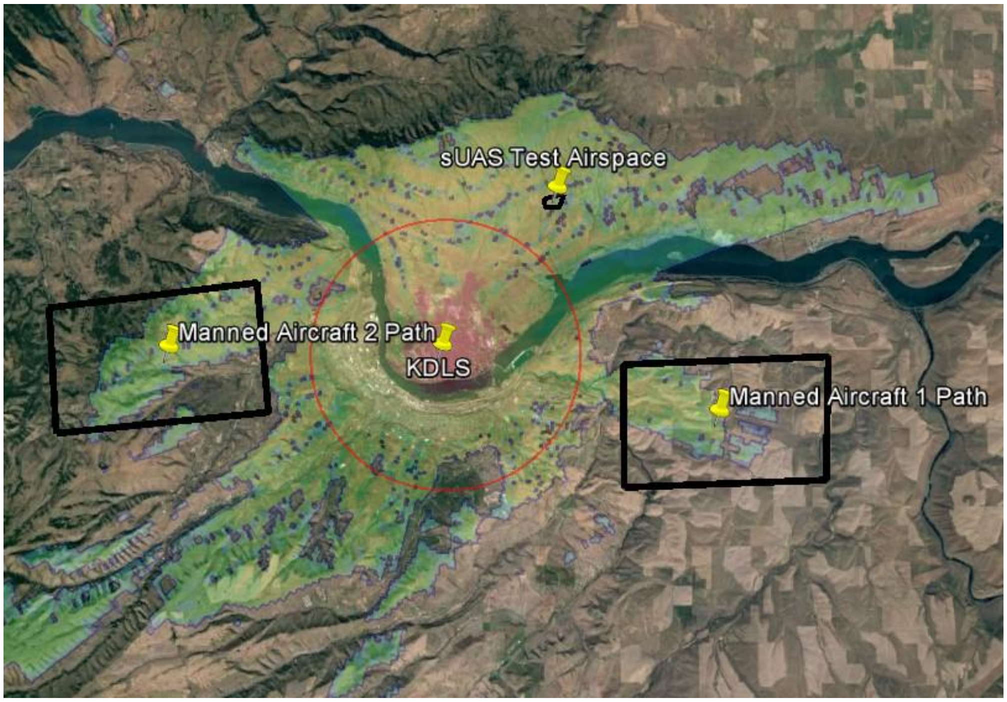


Figure 4.8: KDLS sUAS and manned aircraft airspaces.

flight paths. In keeping with the planned sUAS flight path, the manned aircraft flight paths included climbing and descending portions on the short legs with constant altitude portions on the long legs. Furthermore, the altitudes were selecting in order to comply with an FAA rule governing allowable VFR traffic altitudes. This rule requires that aircraft operating under VFR conditions above 3,000 feet AGL flying a heading between 000 degrees and 179 degrees must fly at odd-thousand foot altitudes plus 500 feet (e.g. 3,500 5,500), and aircraft flying a heading between 180 degrees and 359 degrees must fly at even-thousand foot altitudes plus 500 feet (e.g. 4,500 6,500). This rule ensures adequate separation is maintained between VFR aircraft traveling in separate directions, and the 500 foot addition ensures adequate separation is maintained between VFR and IFR traffic.

For the first manned aircraft flight path to the southeast of the KDLS airfield, the plan was for the aircraft to fly counter-clockwise around the flight path. The long leg on the north side of the path would be flown at 4,500 feet MSL, and the long leg on the south side of the path would be flown at 3,500 feet MSL. The climbing leg was the eastern short leg and the descending leg was the western short leg of the flight path. The second manned aircraft flight path to the west of the KDLS airfield required the aircraft to fly clockwise around the flight path. The long leg on the north side of the flight path would be flown at 5,500 feet MSL, and the long leg on the south side of the flight path would be flown at 6,500 feet MSL. The climbing leg was the eastern short leg and the descending leg was the western short leg.

A risk assessment[22] of the operation taking into account the location of the UAS and participating manned aircraft as well as the general density of non-participating aircraft in the area was performed[23]. It was determined that the spatial separation of the vehicles was sufficient to ensure sufficient safety during the operation. The chosen altitudes ensured that even at their closest, the manned aircraft would have 1,000 feet of altitude separation and 8 NM of lateral separation. Both of the manned aircraft had a minimum of 4 NM of lateral separation from the sUAS flight path and maintained vertical separation of over 1,000 feet. The final flight plan ensured that no unnecessary risks would be taken during the flight testing.

4.1.4 Logistics and Coordination

After all flight test locations and airspaces were finalized, coordination was required to ensure the flight test would proceed as planned on the day of the test. Due to the nature of the flight test with ground crews, sUAS flight crews, and manned aircraft operating in nearby airspace, it was imperative to ensure proper communication was maintained between all involved parties. In order to facilitate this communication, several protocols were developed.

The first problem to be addressed focused on maintaining communication between AFSL ground crew members located at the KDLS airfield and AFSL flight crew members located at the sUAS test site location. During normal AFSL flight testing operations at Meadowbrook Farms, ground station personnel and flight crew personnel are located in the same vicinity, and communications between required personnel are maintained through the use of a conference call on cell phones with hands-free devices. Ground testing conducted in the spring of 2016 indicated that cell phone reception at the sUAS test site would be inconsistent, and therefore other communications options were researched. The plan called for using the cell phone conference call as the primary means of communication between the AFSL ground crew at KDLS and the AFSL flight crew at the flight test location. If cell phone reception proved to be unreliable, the secondary means of communication involved using CB radio units on CB channel 12. Ultimately, cell coverage on the test day allowed for communications via the conference call.

Once a communications solution was devised for AFSL communications between the KDLS ground crew and the flight crew, communications between AFSL personnel and the manned aircraft were considered. During pre-test briefings, the pilots of the manned aircraft were informed of the test plans to include designated test airspace flight paths and altitude blocks. In order to facilitate communication between the manned aircraft pilots and the sUAS ground crew during testing, test crews communicated via the common air-to-ground frequency of 122.9 MHz. The sUAS remote pilot was able to communicate with the manned aircraft pilots from the sUAS ground station by using a Yaesu hand-held air band radio. Additionally, the manned aircraft pilots were able to report flight path progress to the sUAS ground crew, and the pilots were able to communicate with one another in order to avoid any potential aerial conflicts during the flight testing period.

4.2 Test Cards

4.2.1 Planned Test Cards

Considering the initial research goals and the overall scope of the project, several main tests were designed for the KDLS flight demonstration. These tests were intended to demonstrate the aircraft tracking capabilities of the TRAPIS code in GPS-degraded and GPS-denied environments while simultaneously demonstrating autonomous flight capabilities of a commercially-available sUAS operating while using an ADS-B Out transponder.

In order to demonstrate the full range of TRAPIS code and payload capabilities during a one-day flight demonstration, five test cards were created, each corresponding to a separate flight of the aircraft as summarized in Table 4.1. In each case, the planned flight involved the sUAS aircraft flying around the pre-defined rectangular path. Each of the test flights were designed so that the aircraft would navigate the flight path a total of five times before returning to loiter around the home point at an altitude of 250 feet AGL. After becoming established in the loiter, the aircraft would be returned to the ground station for a landing under manual control of the remote pilot in command. All test cards were planned to be conducted while the TacAero manned aircraft were flying in the separate designated flight test locations.

Card Number	**Name**	**Description**
1	GPS Accuracy High	GPS signal not artificially degraded or denied in any manner
2	GPS Accuracy Degraded	GPS signal artificially degraded to NACp of 8
3	GPS Denied	GPS signal artificially denied
4	Intermittent GPS	GPS cycled between normal, degraded and denied operation
5	Evasive Maneuvers	sUAS performs simulated evasive maneuvers with normal GPS operation

Table 4.1: Planned sUAS test cards.

The first test card was designed to demonstrate the normal capabilities of the ADS-B Out transponder while operating in an environment without GPS degradation or denial. The test card called for the aircraft to be autonomously flown around the flight path for a total of five circuits while ensuring that the aircraft GPS signal was not being artificially degraded or denied. During the test, the AFSL ground crew located at the KDLS airport would monitor the TRAPIS software and make sure that both ADS-B and LAMS information were being received for aircraft tracking purposes. Additionally, both of the TacAero manned aircraft would be tracked by the AFSL ground station crew.

The second test card was designed to demonstrate the ability of the TRAPIS software to compensate for sUAS operations in degraded GPS environments by using LAMS information coupled with real-time filtered ADS-B estimates of aircraft position. The aircraft was intended to complete one full circuit of the flight path with a non-degraded GPS signal.

Upon the start of the second circuit, the remote pilot in command was tasked with artificially degrading the GPS information being supplied to the transponder, thereby corrupting the ADS-B position information being reported by the transponder. The GPS degradation was designed to remain in effect for the second, third, and fourth full laps of the flight path. Once the aircraft began the fifth lap of the flight path, the remote pilot in command was once again tasked with changing the GPS information back to a non-degraded state. By degrading the GPS information presented to the transponder during the middle three laps of the flight test, the contrast between normal and degraded GPS environment operations would be shown.

In order to show the operation of the TRAPIS software in GPS-denied environments, the third test card was designed in the same manner as the second test card. During the second, third, and fourth laps of the test, the remote pilot in command was tasked with artificially denying the GPS information provided to the transponder. By artificially denying the GPS signal, the position information displayed to the ground station user through the TRAPIS interface would only contain information from the LAMS system. During the first and final circuits of the planned flight path, the GPS information would not be denied to the transponder so that both the ADS-B and LAMS data streams would be seen by the ground station operator located at the KDLS airport.

Since the second and third test cards focused on testing the TRAPIS aircraft tracking capabilities during GPS-degraded and GPS-denied operations, the fourth test card was designed to test TRAPIS operations during periods of intermittent GPS availability. The method used for testing the payload with simulated intermittent GPS availability was similar to the methods used for the second and third test cards. The first and fifth circuits of the flight path were designed to be flown without GPS degradation or GPS denial. During the middle three circuits of the flight path, the remote pilot in command was tasked with cycling the transponder payload between non-degraded GPS information, artificially degraded GPS information, and artificially denied GPS information. By cycling the GPS information between these three options, the remote pilot in command was able to simulate an aircraft operating with an ADS-B transponder in an intermittent GPS environment.

The final test card was aimed at testing the abilities of the TRAPIS real-time state estimators to keep pace with position estimates for an aircraft that needed to perform evasive maneuvers. Many scenarios exist wherein an aircraft, manned or unmanned, would be required to perform evasive maneuvers in order to avoid an object to include a bird, another aircraft, terrain, or any variety of unplanned obstacle. In order to test for evasive maneuver tracking, the test card called for one circuit of the flight path to be flown autonomously. Once the circuit was complete, the remote pilot in command was tasked with taking manual control of the sUAS and performing evasive maneuvers to include sharp turns, climbs and descents. After the evasive maneuvers were completed, the pilot was tasked with placing the sUAS back into auto mode such that the aircraft would complete one final circuit of the flight path before landing.

Once all of the test cards were generated, the tests were run during initial flight testing excursions at Meadowbrook Farms. It was shown that all test cards could be completed successfully with the test aircraft and the TRAPIS payload, and that the TRAPIS user interface behaved as expected. These initial tests proved that all test cards were feasible and could be performed using the same type of rectangular flight path during the final KDLS flight demonstration.

4.2.2 Modified Test Cards

Flight test plans are drafted with the hope that test day conditions will allow all test cards to be completed as planned. In reality, unforeseen factors often prevent flight tests from proceeding according to the exact test plan, and this was the case with the culminating JCATI flight test. On both September 22nd and September 23rd, weather conditions at the intended test site prevented the planned test cards from being completed and restricted flight options to manually-controlled flights in which the test sUAS would not have been able to autonomously complete the planned flight path. Additionally, unforeseen complications with the primary transponder unit and line-of-sight issues at the flight test site required alterations to the planned test cards. These challenges are described in greater detail in Chapter 7 of this report.

During the first day of flight testing on September 22nd, it was determined that the original flight testing airspace would not suffice for the required flight tests. The LAMS system required line-of-sight with the test vehicle in order to successfully track the sUAS, and from the mobile GCS there was no line-of-sight between the sUAS and the LAMS station at the KDLS airfield. Accordingly, the aircraft needed to be launched and set on the planned flight path at the appropriate test altitudes before line-of-sight with the LAMS station could be achieved. In addition to the LAMS station, the ADS-B In receiver required line-of-sight with the transponder antenna in order to receive updates on aircraft position. This requirement further complicated matters with the use of the original test location, since both LAMS and ADS-B tracking required line-of-sight operation of the sUAS.

Before it was determined that the original transponder unit was not sending interrogation responses with the proper pulse width, it was concluded that the issues with the sUAS flight testing must have stemmed from the lack of line-of-sight between the GCS at the test site and the ground station at the KDLS airport. Subsequent ground tests with a flight team member holding the test sUAS and walking near the test site showed improved transponder performance when the aircraft was moved to a hill 500 yards east of the original test location. The location of this hill in relation to the original GCS site in the proposed test airspace is shown in Figure 4.9. From the top of the hill there was line-of-sight between the KDLS ground station and the GCS, which allowed test personnel to troubleshoot connection issues between the sUAS, the LAMS, and the ADS-B In receiver before launching the aircraft.

After analyzing the results of the September 22nd ground tests at the original and alternate locations, it was determined that further tests on September 23rd should be conducted at the alternate test location on top of the hill. Strong winds continued to affect flight operations on the second day of testing, and COA requirements would have prevented the AFSL flight crew from maintaining line-of-sight with the sUAS during autonomous flight operations using the original flight path. In light of these factors, the plan changed from

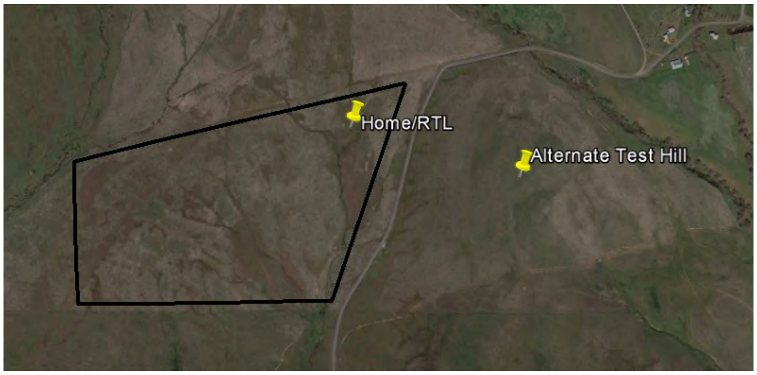


Figure 4.9: KDLS alternate sUAS test location.

conducting autonomous flights around the original flight path to conducted manual flights near the alternate GCS location.

Since the test sUAS had never been flown in sustained 20-30 mph winds, there was potential for the aircraft to crash during flight testing. In order to ensure that all required data would be gathered during a successful flight, it was determined that all three GPS states would be tested during a successful flight. During manual operations, the remote pilot in command would fly the sUAS for several manually-controlled circuits around the test location with no artificial GPS degradation or denial before artificially degrading the GPS signal provided to the transponder for several additional circuits. The pilot would then artificially deny the GPS signal for several circuits before returning the GPS information to its non-degraded state for the final circuits of the test. This adjusted test plan was followed for the flight tests that were conducted from the alternate GCS location on September 23rd. Although the modified tests did not allow for each of the GPS degradation states to be tested on separate flights, it allowed the flight crew to test all required functionality while minimizing the risk of successive test flights in an extremely challenging flight environment.

Chapter 5

ESTIMATION AND FUSION

5.1 Estimation Algorithms

In order to accurately estimate and fuse vehicle flight paths from ADS-B and LAMS information sources in varying states of degradation or denial, several estimation algorithms were developed. These estimation algorithms ranged in complexity and provided a multi-faceted approach to accurate estimation of vehicle flight paths. At the lowest level, the estimation algorithms passed unaltered position data through the TRAPIS framework to the user interface, and at the highest level the estimation algorithms accounted for position errors associated with ADS-B and LAMS TRAPIS information packets in real-time. The three estimators developed for this project allowed for a build-up approach to accurate vehicle position estimation, and provided TRAPIS users with a variety of estimation tools to choose from.

In order to standardize the implementation of the estimators within the overall TRAPIS code structure, a framework was created under which all estimation algorithms were defined. This framework consisted of a C# interface object defined as *IEstimator*. Within this interface several methods were created, the specific members of which were required to be defined separately for each of the estimators. The interface framework ensured that a common implementation was used by all of the estimation algorithms for integration within TRAPIS.

Based on the requirements of the interface, each estimation algorithm had to have access to a list of ADS-B or LAMS observations, *Observations*, from which the estimation algorithm could gather vehicle state information to be used in the state estimation. In addition to this list of observations, each estimation algorithm was required to have three primary methods defined. The first of these methods, *AddObservation*, was used to gather the most recent ADS-B or LAMS observation and add it to the list of observations for the estimator. The second, *ComputeEstimate*, was used to produce the state estimate from the ADS-B or LAMS observations associated with the estimator. The third, *DeepCopy*, returned a copy of the estimator object. These methods were required to be defined for all estimation algorithms at a minimum in order to ensure standardization of estimation methods to be called by the fusion algorithms. Further discussion of the specifics of each estimation algorithm will focus on the processes associated with the *ComputeEstimate* method, as the *AddObservation* and *DeepCopy* methods remained the same for all estimation algorithms and only involved trivial functionality.

5.1.1 DoNothingEstimator

At the lowest level, estimation associated with the TRAPIS software relied on unaltered ADS-B and LAMS position information being passed directly to the TRAPIS user interface. In accordance with this description of the basic information handoff that occurred for the lowest-level estimator, this simple estimator was called the *DoNothingEstimator*. Although the *DoNothingEstimator* did not alter the position information being provided to the TRAPIS interface, it provided the basic building block from which additional estimators were developed. Additionally, assuming that all ADS-B and LAMS information being ingested by the TRAPIS software is not being degraded or denied, then the *DoNothingEstimator* avoids unnecessary filtering of pre-filtered ADS-B and LAMS data.

As detailed in the overall description of the estimation algorithms, the required methods were defined for the *DoNothingEstimator* to ensure proper implementation of the estimation algorithm within the overall TRAPIS framework. Since the estimator was designed to be a simple pass-through of state observations from the ADS-B and LAMS data streams to the fusion algorithms, the *ComputeEstimate* method definition associated with the estimation algorithm was similarly simple. Due to the fact that the *DoNothingEstimator* uses the current ADS-B and LAMS position information provided to the TRAPIS interface in the data streams, no buffer of positions is maintained. Accordingly, the *ComputeEstimate* method gathers this most recent observation from the *Observations* list and returns this observation as the computed estimate in the form of a *TRAPISPacket*.

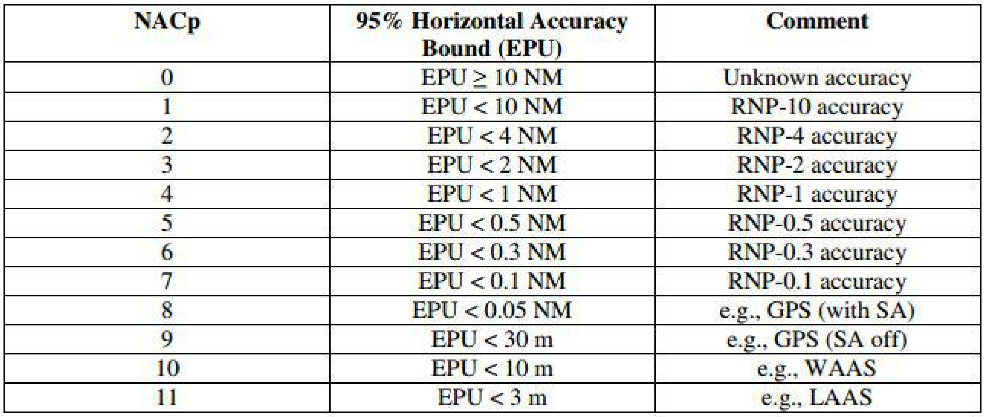
The *DoNothingEstimator* was primarily used as an initial test of estimation algorithm implementation within the broader TRAPIS framework. Since the estimator did not involve any computations or actual state estimation, it was not included in the final version of the TRAPIS software used during the culminating flight demonstration at KDLS.

5.1.2 Kalman Filter Estimator

Once the *DoNothingEstimator* was completed, two additional estimators were developed. The additional estimators were designed in order to incrementally increase the complexity of the position estimates provided to the data fusion algorithms. During initial outlining and planning, it was determined that the advanced position estimation algorithms should include filters designed to reduce the known errors associated with position measurements in order to obtain consistent, accurate estimates of aircraft positions.

The first of these two estimators, the *KalmanFilterEstimator*, was designed to reduce the error associated with position estimates through the use of basic filtering with assumed position errors. Based on several factors including the number of available GPS satellites, the orientation of said satellites, the orientation of the GPS antenna on the receiving unit, and others, the accuracy of GPS position measurements can change significantly over the course of a measurement period. In ADS-B applications, this measurement error is reported as a NACp value, for which there are corresponding position errors and altitude errors. A table of these NACp values and their associated position errors is shown in Figure 5.1 [25].

From the information contained in Figure 5.1 it can be seen that for each NACp value there is an associated EPU, Estimated Position Uncertainty, which corresponds to the horizontal position error for a 95% horizontal position accuracy bound. The range of available NACp values gives a wide range of position errors for GPS position measurement accuracy. Reported NACp values in ADS-B Out packets fall in the range from 7 to 11 during normal



NACp	95% Horizontal Accuracy Bound (EPU)	Comment
0	EPU ≥ 10 NM	Unknown accuracy
1	EPU < 10 NM	RNP-10 accuracy
2	EPU < 4 NM	RNP-4 accuracy
3	EPU < 2 NM	RNP-2 accuracy
4	EPU < 1 NM	RNP-1 accuracy
5	EPU < 0.5 NM	RNP-0.5 accuracy
6	EPU < 0.3 NM	RNP-0.3 accuracy
7	EPU < 0.1 NM	RNP-0.1 accuracy
8	EPU < 0.05 NM	e.g., GPS (with SA)
9	EPU < 30 m	e.g., GPS (SA off)
10	EPU < 10 m	e.g., WAAS
11	EPU < 3 m	e.g., LAAS

Figure 5.1: NACp values and associated position errors [25].

operations, assuming there is no degradation or jamming of the GPS signal being received by the ADS-B transceiver unit. For the purposes of this project however, the entire range of NACp values was acceptable for use since the GPS signal was allowed to be degraded or denied.

Once the NACp table shown in Figure 5.1 was found and initial analysis of real ADS-B Out packets was conducted, an initial choice of horizontal and vertical positions errors was required for the *KalmanFilterEstimator* position estimation algorithm. It was determined that for the requirements of the *KalmanFilterEstimator*, a NACp value of 9 would be used, corresponding to a horizontal position error of 10 meters. In addition to the assumed horizontal position error, the vertical position error was assumed to be 7.5 meters. This vertical position error was chosen as a result of the altitude reporting capabilities of the Sagetech XPS-TR transponder unit. The transponder was only able to resolve pressure altitude differences of 25 feet, and as a result the vertical position error was selected to be 7.5 meters which corresponds to approximately 25 feet.

The same horizontal and vertical position errors that were assumed for the ADS-B position estimates were also assumed for the LAMS position estimates. Although the LAMS systems is subject to position errors that are dependent on the range of tracked aircraft from the LAMS system, all sUAS and manned aircraft testing required for the culminating flight test was performed within 10 nautical miles of the LAMS equipment site at the KDLS airfield. Furthermore, the horizontal position accuracy of the LAMS system manifests itself in both range and bearing errors. Although these horizontal position errors are not directly related to those for the GPS information used in the ADS-B packets, it was assumed that the errors were similar for the test applications required of the tracking algorithms. Since the maximum reliable tracking range of the LAMS system is a 60 nautical mile radius from the LAMS tracking station, but all flight operations were conducted within a 10 nautical mile radius from the LAMS station, it was determined that 10 meters was a reasonable assumption for the LAMS horizontal position errors. Additionally, the *KalmanFilterEstimator* only served as an interim estimation algorithm and was replaced by the *DynamicKalmanFilter-Estimator* in the final iteration of the TRAPIS software, so any errors associated with the assumptions made for the *KalmanFilterEstimator* were taken care of in the final TRAPIS product.

As evidenced by the name associated with the filter, the *KalmanFilterEstimator* was designed to function as a basic discrete time Kalman filter. The prediction and update equations associated with the Kalman filter are shown in general form in Equations 5.1 through 5.5 [34]. Since there were no systems inputs that directly affected the system state, no terms for an input vector were included in the prediction and update equations associated with the *KalmanFilterEstimator*.

$$x_{predicted} = A * x_{n-1} \tag{5.1}$$

$$P_{predicted} = A * P_{n-1} * A^T + F * Q * F^T \tag{5.2}$$

$$K = P_{predicted} * H^T * \Big(H * P_{predicted} * H^T + R\Big)^{-1} \tag{5.3}$$

$$x_n = x_{predicted} + K * \Big(Y - H * x_{predicted}\Big) \tag{5.4}$$

$$P_n = \Big(I - K * H\Big) * P_{predicted} \tag{5.5}$$

Using the discrete time Kalman Filter equations detailed above for both the ADS-B and LAMS information being supplied to the TRAPIS position estimation framework required a standard reference from which to compute position changes during a flight scenario. Without a standard framework and geographical reference for the position measurements made using both the LAMS and ADS-B systems, there would be no way to accurately compare the position estimates generated using the two methods. Based on this requirement, all vehicle positions, velocities, and associated errors were transformed into a local coordinate system defined by the AFSL as the *UW* coordinate frame. In essence, the methods associated with the *UW* coordinate system transform coordinates and velocities associated with a vehicle system into a North, East, Down frame in order to allow for ease of calculation. Once all required calculations are performed, the *UW* coordinates can be converted back into known distances and velocities within the original frame.

An additional benefit of the *UW* coordinate system is that it allows for direct application of velocities over specified time periods culminating in changes to the latitude and longitude positions of an aircraft. The coordinate transformations involved in reconciling position and velocity changes over changes in latitudes and longitudes are numerous and can be rather complex depending on the distances and magnetic travel directions involved. By using the *UW* coordinate system, these conversions are performed directly by the code base, and it is not left to the user to define required coordinate transformations each time new code is written.

Regardless of whether a new *KalmanFilterEstimator* object was associated with ADS-B or LAMS information, the position and velocity information associated with the corresponding *TRAPISPacket* was immediately converted into the *UW* coordinate system. Once the distances, velocities, and associated errors had been transformed into the new coordinate system, the state vector was generated according to the definition shown in Equation 5.6.

$$x = \begin{bmatrix} EastPosition(m) \\ EastVelocity(m/s) \\ NorthPosition(m) \\ NorthVelocity(m/s) \\ DownPosition(m) \\ DownVelocity(m/s) \end{bmatrix} \tag{5.6}$$

Based on the requirements of the *UW* North, East, Down coordinate frame, each of the entries included in the state vector was an element of another vector associated with the position and velocity information included within the TRAPIS packets. The *UWNEDVector* operations require that the positions associated with a vehicle are represented as vectors from an origin to the current vehicle position. Based on these requirements, and based on the nature of the discrete time Kalman filter wherein the state vector is updated at each time step, the aircraft position had to be defined as a vector for each new time step. Accordingly, once the new state vector was calculated at the given time step, the previous position estimate was used as the new origin, and the positions from the next *TRAPISPacket* to enter the filter were used as the positions for the new position vector, the origin of which was the position estimate from the previous time step. Based on the requirement that the aircraft positions had to be represented as vectors in the *UW* North, East, Down frame, the *KalmanFilterEstimator* needed to have at least three *TRAPISPacket* observations added to the *Observations* list before position estimates could be generated.

In addition to the state vector, the required prediction and update matrices had to be defined for the Kalman filter to work. Since the *KalmanFilterEstimator* is a discrete time Kalman filter, the time step associated with the system had to be computed and used in the matrices associated with the Kalman filter equations. Due to the nature of the TRAPIS packets generated from the ADS-B and LAMS data streams, the time stamps associated with the packets are not generated at a constant rate. As such, the time between when packets are received in the estimation algorithm is different for each new packet. In order to compensate for this, the time step associated with the Kalman filter is changed each time a new observation enters the estimation algorithm. This time step is calculated as the difference between the time the new packet was received and the time the previous packet was received.

The measurement error for the *KalmanFilterEstimator* was set equal to the assumed horizontal position error of 10 meters for the North and East directions, and 7.5 meters for the Down direction in the *UW* coordinate frame. These assumed position errors were used to directly model the measurement noise associated with the system and form the measurement covariance matrix. Since the coordinate system required transformation of the positions and velocities to the local *UW* coordinate frame, the process noise associated with the system was more difficult to quantify. After many simulations were completed to analyze the effects of changes to the process noise covariance matrix, a choice was made to develop the process noise covariance matrix using the errors associated with the position and velocity measurements of the system. The matrices associated with the discrete time Kalman filter algorithm in the *KalmanFilterEstimator* are detailed in Equations 5.7 through 5.12. The time step calculated between successive observations is shown as dt, and the horizontal and vertical position errors are shown by σ_x and σ_y, respectively.

$$A = \begin{bmatrix} 1 & dt & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & dt & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & dt \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.7}$$

$$F = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.8}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.9}$$

$$P = \begin{bmatrix} \sigma_x & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_x^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_y & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_y^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_z & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_z^2 \end{bmatrix} \tag{5.10}$$

$$Q = \begin{bmatrix} \sigma_x & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_x^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_y & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_y^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_z & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_z^2 \end{bmatrix} \tag{5.11}$$

$$R = \begin{bmatrix} \sigma_x^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{\sigma_x^2}{10} & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_y^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\sigma_y^2}{10} & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_z^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{\sigma_z^2}{10} \end{bmatrix} \tag{5.12}$$

Using these matrices and the discrete time Kalman filter equations detailed in Equations 5.1 through 5.5, the positions and velocities of the tracked vehicle were computed at each new time step. Once the Kalman filter prediction and update equations had been run for each time step, the final velocities in the North, East, Down coordinate system were gathered directly from the output vector. The final positions were determined by offsetting the defined origin by the final North, East, Down position vectors contained within the Kalman filter output state vector.

5.1.3 DynamicKalmanFilterEstimator

After the *KalmanFilterEstimator* was completed, the third and final estimation algorithm, the *DynamicKalmanFilterEstimator*, was developed. In a similar fashion to the *KalmanFilterEstimator*, the *DynamicKalmanFilterEstimator* is a discrete time Kalman filter which uses the same prediction and update equations detailed in Equations 5.1 through 5.5. Unlike the *KalmanFilterEstimator*, the *DynamicKalmanFilterEstimator* uses the actual horizontal and vertical position errors associated with the information contained within the TRAPIS packets to update the required Kalman Filter matrices at each new time step. While the *KalmanFilterEstimator* assumed that the horizontal position error was constant at 10 meters and the vertical position error was constant at 7.5 meters, the *DynamicKalmanFilterEstimator* uses the appropriate position error associated with the TRAPIS packets for either the ADS-B or the LAMS data stream, thereby ensuring that the appropriate position error is used to generate the position estimates for the next time step.

After simulations were conducted using all three of the estimation algorithms, it was shown that the *DynamicKalmanFilterEstimator* provided the best estimates of aircraft position regardless of the integrity of the ADS-B and LAMS data streams. As a result of these simulation results, the *DynamicKalmanFilterEstimator* was selected for use with the final iteration of the TRAPIS software package, and this estimation algorithm was used for all subsequent simulation and testing, to include the final flight demonstration conducted in September of 2016.

5.2 Fusion Algorithms

Once the three estimators were created for the TRAPIS interface, three data fusion algorithms were created as well. In similar fashion to the estimation algorithms, these data fusion algorithms were created with varying levels of complexity in order to provide a multi-tiered approach to fusion of vehicle position estimates. At the lowest level, the fusion of vehicle position estimates operates as a switch between the ADS-B position estimates and the LAMS position estimates, allowing either the ADS-B estimates or the LAMS estimates to be used for the fused position estimate. At the highest level, the fusion algorithms determine the weighted average of ADS-B and LAMS position estimates based on the reported error associated with the ADS-B and LAMS data streams. Ultimately, the three data fusion algorithms complement the three estimation algorithms and allow for varying levels of complexity for position estimation, fusion, and reporting in the TRAPIS software interface.

5.2.1 SimpleFuser

At the lowest level, the fusion algorithms associated with the TRAPIS interface offer a selection of either the ADS-B position estimates or the LAMS position estimates. In a similar fashion to the *DoNothingEstimator* which passed either the unmodified ADS-B or LAMS data stream through to the fusion algorithm, the *SimpleFuser* passed either the GPS or LAMS estimate through as the fused estimate seen by the user through the TRAPIS user interface.

The *SimpleFuser* works with any of the three possible position estimation algorithms to provide a fused position estimate to the user of the TRAPIS software. Once an estimate is generated for the ADS-B or LAMS data being supplied to the corresponding estimation algorithm, the estimate is passed to the *DataFusionController*. If the *SimpleFuser* is being used, then the TRAPIS user can select which estimate is desired, namely the ADS-B or LAMS estimate. The estimates from the chosen data stream are thereafter reported to the TRAPIS user through the associated user interface. By viewing the raw data streams being consumed by the TRAPIS software in real-time, the user can determine which of the two data streams should be trusted over the other. Using the *SimpleFuser*, the user can then select which data stream should be used as the fused estimate. Although the position information provided to the fusion algorithm within the two separate data streams is not combined in any manner, the *SimpleFuser* is well-suited for situations in which one of the two data streams is unaltered while the other data stream is being degraded or denied.

5.2.2 WeightedFuser

After the *SimpleFuser* algorithm was developed, two additional fusion algorithms were developed, the *WeightedFuser* and the *KalmanFuser*. The first of these two fusion algorithms, the *WeightedFuser*, provides increased estimate fusion functionality over the *SimpleFuser* by averaging the position estimates provided to the TRAPIS software by the ADS-B and LAMS data streams. In order to accomplish this weighted averaging, the *WeightedFuser* computes the weighted geographic midpoint of the most recent positions provided to the TRAPIS software by the ADS-B and the LAMS data streams by means of weighted averaging methods. By using a weighted average as opposed to a direct calculation of the unweighted geographic midpoint, the error associated with the ADS-B and LAMS packets provided to the TRAPIS software can be included in the calculation. Depending on which of the two data streams is providing more accurate position information, the geographic midpoint can be corrected appropriately.

In order to implement the weighted averaging required for operation of the fusion algorithm, the *WeightedFuser* accepts two estimation algorithm objects as inputs. If no estimation objects are specified as inputs then the code automatically generates two *DynamicKalmanFilterEstimator* objects. The two associated estimators correspond to an estimator for the ADS-B data stream and an estimator for the LAMS data stream, respectively. Each time new observations are added to the estimators, the *ComputeFusedEstimate* method is called by the TRAPIS software, thereby initiating the averaging required to provide a fused estimate of aircraft position to the TRAPIS user. Once this method is called, the most recent ADS-B and LAMS estimates are provided to the fusion algorithm along with the associated position errors.

After the most recent state estimates are received by the *WeightedFuser*, the weighted geographic midpoint is computed. In order to compute this midpoint, both position estimates are first converted to a local Cartesian coordinate system. The equations required for this conversion are shown in Equations 5.13 through 5.15, where *lat* corresponds to the latitude of the estimate, and *lon* corresponds to the longitude of the estimate [15].

$$X = \cos(lat) * \cos(lon) \tag{5.13}$$

$$Y = \cos(lat) * \sin(lon) \tag{5.14}$$

$$Z = \sin(lat) \tag{5.15}$$

Once these Cartesian coordinates are calculated for both the ADS-B and LAMS estimates, the weighting factors are determined by using the planar standard deviations reported in the TRAPIS packets associated with the most recent observations. These weighting factors are determined by Equation 5.16 where σ_{planar} represents the planar standard deviation of the corresponding ADS-B or LAMS estimate. Using the inverse ensures that the measurements with larger associated planar standard deviations carry less weight in the final fused estimate.

$$w = \frac{1}{\sigma_{planar}} \tag{5.16}$$

Using the Cartesian coordinates computed for both the ADS-B and the LAMS estimates along with the weighting factors determined by the planar standard deviations associated with each of the estimates, the weighted Cartesian coordinates are determined using Equations 5.17 through 5.19. The subscripts on the variables indicate the Cartesian coordinates and weighting factors associated with the ADS-B and LAMS estimates, respectively.

$$X_{fused} = \frac{\Big(X_{ADSB} * w_{ADSB} + X_{LAMS} * w_{LAMS}\Big)}{\Big(w_{ADSB} + w_{LAMS}\Big)} \tag{5.17}$$

$$Y_{fused} = \frac{\Big(Y_{ADSB} * w_{ADSB} + Y_{LAMS} * w_{LAMS}\Big)}{\Big(w_{ADSB} + w_{LAMS}\Big)} \tag{5.18}$$

$$Z_{fused} = \frac{\Big(Z_{ADSB} * w_{ADSB} + Z_{LAMS} * w_{LAMS}\Big)}{\Big(w_{ADSB} + w_{LAMS}\Big)} \tag{5.19}$$

After the fused Cartesian positions are calculated, the fused coordinates are converted back into a fused latitude and longitude estimate using Equations 5.20 through 5.22.

$$Lon_{fused} = \text{atan2}(Y_{fused}, Xfused) \tag{5.20}$$

$$Hyp = \sqrt{X_{fused}^2 + Y_{fused}^2} \tag{5.21}$$

$$Lat_{fused} = \text{atan2}(Z_{fused}, Hyp) \tag{5.22}$$

The latitude and longitude coordinates obtained from Equations 5.20 through 5.22 result in the weighted geographic midpoint of the ADS-B and LAMS position estimates. Once these coordinates are calculated, the weighted average of the aircraft velocity vector in terms of North, East, Down components is determined using Equations 5.17 through 5.19. The same planar deviation weighting factors are used for the fused velocity calculations. Once the fused North, East, and Down velocity components are determined, they are placed into a *UW* velocity vector and included in the *TRAPISPacket* generated for the fused estimate.

The weighted average of the aircraft altitude is calculated by standard methods using Equations 5.23 and 5.24. In these equations, the subscripted altitudes correspond to the altitudes reported in the most recent ADS-B and LAMS observations provided to the fusion algorithm, and the variable σ_{alt} corresponds to the reported altitude standard deviation in the most recent ADS-B or LAMS observation.

$$w_{Alt} = \frac{1}{\sigma_{alt}} \tag{5.23}$$

$$Alt_{fused} = \frac{\Big(Alt_{ADSB} * w_{Alt:ADSB} + Alt_{LAMS} * w_{Alt:LAMS}\Big)}{\Big(w_{Alt:ADSB} + w_{Alt:LAMS}\Big)} \tag{5.24}$$

The fused planar and altitude standard deviations are calculated using a direct average, and these fused standard deviations are included in the appropriate fields of the *TRAPISPacket* associated with the fused estimate. Each time new observations are added to both of the estimators associated with the fusion algorithm, the *WeightedFuser* computes a new fused estimate which is then provided to the TRAPIS user interface to be viewed by the user. The *WeightedFuser* was used for preliminary testing of the weighted geometric midpoint calculations before the third fusion algorithm was developed.

5.2.3 KalmanFuser

Once the *WeightedFuser* was completed, the *KalmanFuser* was developed. The *KalmanFuser* was designed to further streamline the fusion of ADS-B and LAMS estimates by directly weighting the estimated state vectors associated with both estimation algorithms.

In order to perform this data fusion, Equation 5.25 was used as detailed in [7]. The equation provides a means of finding the weighted average of the state vectors, X, associated with two Kalman filters by means of their error covariance matrices, P.

$$X_{fused} = \frac{\Big(X_{ADSB}/P_{ADSB} + X_{LAMS}/P_{LAMS}\Big)}{\Big(1/P_{ADSB} + 1/P_{LAMS}\Big)} \tag{5.25}$$

Equation 5.25 assumes that the Kalman filters used are associated with redundant streams of information, namely sensors that are providing the same state measurements to the system. For this application, since the ADS-B and LAMS data streams provide the same vehicle information to the TRAPIS software by means of TRAPIS packets, the equation proved useful.

If this direct weighting of the estimated state vectors failed, the *KalmanFuser* used the same methods as the *WeightedFuser* to compute an appropriate fused estimate of the vehicle position and velocity. In the event that one of the data streams was no longer providing information to the fusion algorithm, in the event of GPS-denial for example, the *KalmanFuser* was designed to provide the most recent estimate of the non-denied data stream as the fused estimate. Upon re-acquisition of the denied signal, the *KalmanFuser* would once again weight the positions and velocities based on the standard deviations associated with the most recent estimates from both data streams. Accordingly, the *KalmanFuser* was used with the *DynamicKalmanFilterEstimator* for the final iteration of the TRAPIS software to provide the most appropriately-fused vehicle data to the user.

Chapter 6

SIMULATION AND INITIAL TESTING RESULTS

6.1 Simulation Results

Before flight test data were gathered with the ADS-B payload flown on an AFSL sUAS, a variety of simulations were run. These simulations were designed to ensure that all TRAPIS code worked as planned, specifically as it related to the implementation of the estimation and fusion algorithms. Although the final TRAPIS software only made use of the *DynamicKalmanFilterEstimator* estimation algorithm and the *KalmanFuser* data fusion algorithm, all of the estimation and fusion algorithms were tested in simulation. Through simulation, it was shown that all estimation and fusion algorithms performed as desired within the TRAPIS software framework.

6.1.1 Simulation Scenario

In order to accurately model the flight test that was planned as the culminating event of the TRAPIS research, a simulation scenario was developed in which three aircraft were flying in geographically-separate airspaces around the KDLS airfield. These airspaces were based on the initial flight test design, before additional factors and land-use agreements prevented the original test airspaces from being used as-planned. The original test design involved an AFSL sUAS carrying the TRAPIS payload flying in an airspace 3.5 nautical miles North-Northeast of the KDLS airfield, with two manned aircraft flying 5.2 nautical miles to the North and 5.2 nautical miles to the East of the KDLS airfield, respectively. The locations of these original test areas are shown in Figure 6.1.

In Figure 6.1 the test airspaces are marked by the black rectangles with yellow pin identifiers. These approximate locations were used for the simulated aircraft in order to

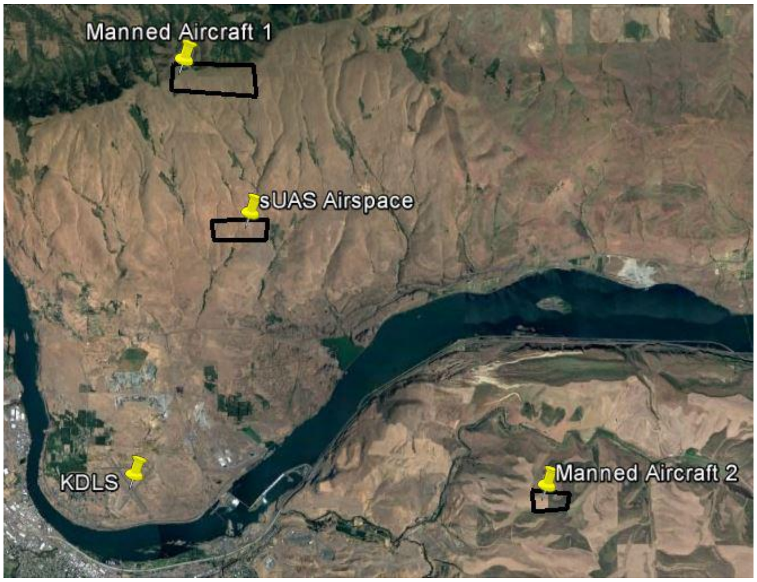


Figure 6.1: TRAPIS simulation scenario airspaces.

closely match the planned flight test airspaces and aircraft flight paths.

After the simulation aircraft and routes were created, the simulations were completed in order to test all of the estimation and fusion algorithms. In order to ensure that the estimation and fusion algorithms worked as designed from the lowest to highest levels, a build-up testing approach was used. By using this build-up approach, low-level code problems could be identified in the less-complex algorithms and fixed before continued testing with the advanced algorithms. Based on this desired build-up method, the simulations were conducted in accordance with the framework shown in Table 6.1. Using this run structure all estimation and fusion algorithms were tested with one another. Each of the runs were conducted multiple times, with variation in the length and scope of ADS-B and LAMS data stream degradation and denial. Simulation results showed that all estimation and fusion algorithms worked as-desired, even in the presence of ADS-B and LAMS data stream degradation or denial.

In order to accurately summarize the simulation results and the performance of each of the algorithms, several of the simulation runs are summarized in the following sections as

Pair Number	**Estimation Algorithm**	**Fusion Algorithm**
1	DoNothingEstimator	SimpleFuser
2	DoNothingEstimator	WeightedFuser
3	DoNothingEstimator	KalmanFuser
4	KalmanFilterEstimator	SimpleFuser
5	KalmanFilterEstimator	WeightedFuser
6	KalmanFilterEstimator	KalmanFuser
7	DynamicKalmanFilterEstimator	SimpleFuser
8	DynamicKalmanFilterEstimator	WeightedFuser
9	DynamicKalmanFilterEstimator	KalmanFuser

Table 6.1: Simulation run matrix.

detailed in Table 6.2 and Table 6.3. For the runs conducted using the *SimpleFuser*, the LAMS data stream was not degraded. This was done in order to accurately simulate the LAMS data stream being chosen over a degraded or denied ADS-B data stream. Furthermore, for all ADS-B denied operations, the NACp value was set to 8, with corresponding standard deviations applied to the TRAPIS packets as detailed in Figure 5.1.

Run Number	**Estimation Algorithm**	**Fusion Algorithm**
1	DoNothingEstimator	SimpleFuser
2	KalmanFilterEstimator	WeightedFuser
3	DynamicKalmanFilterEstimator	KalmanFuser
4	DynamicKalmanFilterEstimator	KalmanFuser

Table 6.2: Abbreviated simulation run matrix.

Run Number	**ADS-B Degraded**	**LAMS Degraded**	**ADS-B Denied**
1	Yes	No	No
2	Yes	Yes	No
3	Yes	Yes	No
4	Yes	Yes	Yes

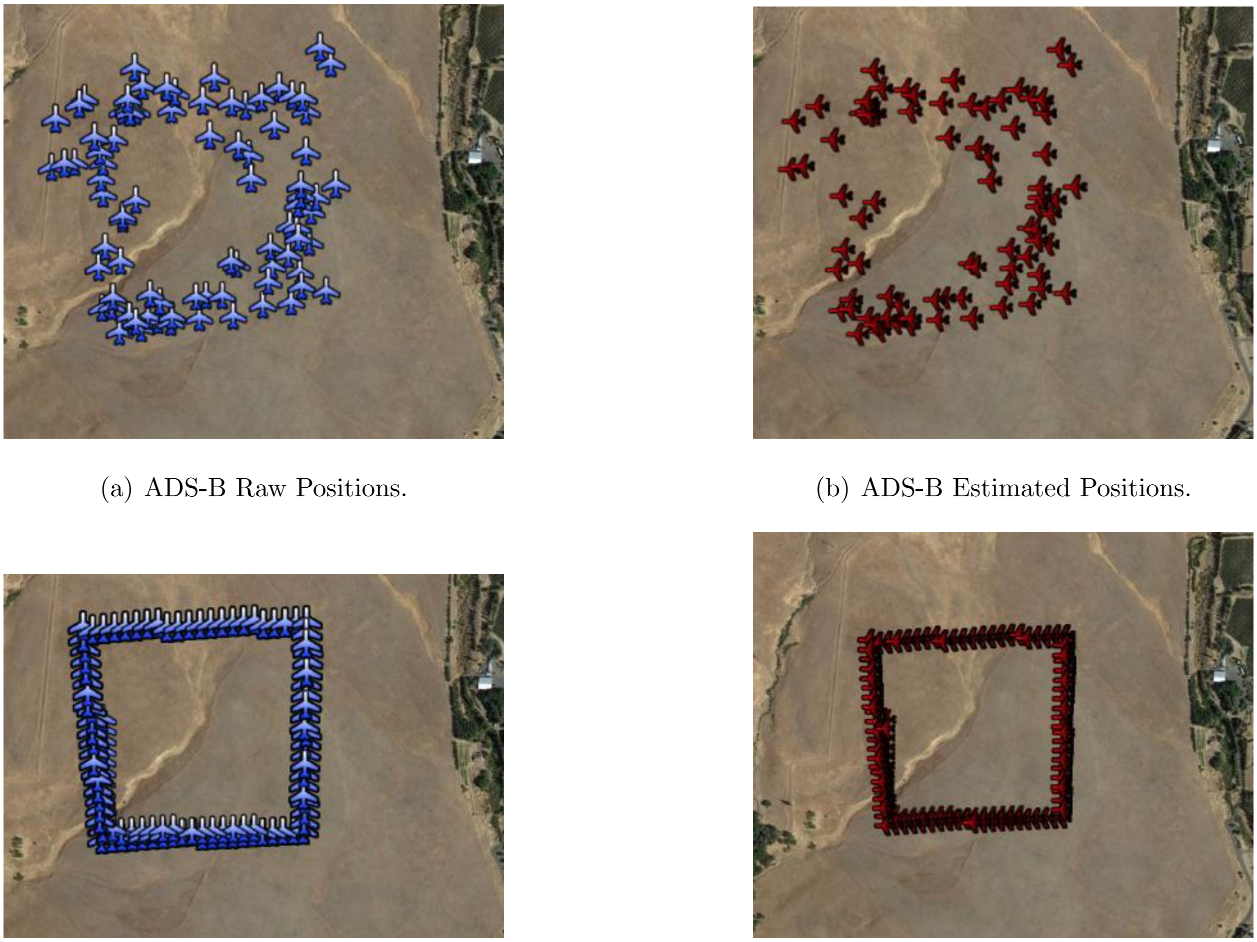
Table 6.3: Simulation ADS-B and LAMS degradation states.

6.1.2 DoNothingEstimator and SimpleFuser

At the lowest-level, position estimation involved the use of the raw, unaltered ADS-B and LAMS data streams by means of the *DoNothingEstimator*. Since the *DoNothingEstimator* did not alter the position information gathered from the data streams in any manner, the scope of acceptable applications was necessarily limited. For the purposes of simulation, the only application of the *DoNothingEstimator* that provided realistic results was one in which one of the two data streams was unaffected by degradation or denial. In order to create such a situation, the simulated ADS-B data stream was assumed to be degraded from an optimal NACp value of 10 to a NACp value of 8. The ADS-B data stream was degraded during the entire simulation time span of 350 seconds, but the LAMS data stream was assumed to be unaltered, thereby providing the TRAPIS user with true aircraft positions.

In order to check the performance of the *DoNothingEstimator*, two .kml files were generated at the end of the simulation. These .kml files contained the information from the raw ADS-B and LAMS data streams as well as the information from the estimated ADS-B and LAMS data streams. Once these .kml files were generated, they were opened in Google Earth and the data from the raw streams were compared to the data from the estimated streams. For the *DoNothingEstimator* the estimated data were expected to be the same as the raw data, a result which was confirmed by the simulation. The data associated with the ADS-B and LAMS streams for the simulated sUAS are shown in Figure 6.2. The raw position data are shown by blue aircraft markers, while the estimated position data are shown by red aircraft markers.

From the figure, it can be seen that the *DoNothingEstimator* performed as anticipated, where the estimated positions are equivalent to the raw positions for both the ADS-B and LAMS data streams. Figures 6.2(a) and 6.2(b) show the raw and estimated position information corresponding to the ADS-B data, while Figures 6.2(c) and 6.2(b) show the same information for the LAMS data during one circuit of the planned sUAS flight path. The simulated sUAS traveled in a counter-clockwise direction, and began its circuit in the northwest



(a) ADS-B Raw Positions.

(b) ADS-B Estimated Positions.

(c) LAMS Raw Positions.

(d) LAMS Estimated Positions.

Figure 6.2: *DoNothingEstimator* simulated ADS-B and LAMS raw and estimated positions.

corner of the flight path. Since the estimation algorithm did not start providing estimates until at least three TRAPIS Packets were received, it can be seen that the initial raw positions provided to the TRAPIS software were not associated with corresponding estimated positions.

While the *DoNothingEstimator* was being used to generate the estimated positions associated with the ADS-B and LAMS data streams, the *SimpleFuser* was being used to return the most accurate of the two estimated streams. In the case of this initial simulation, the

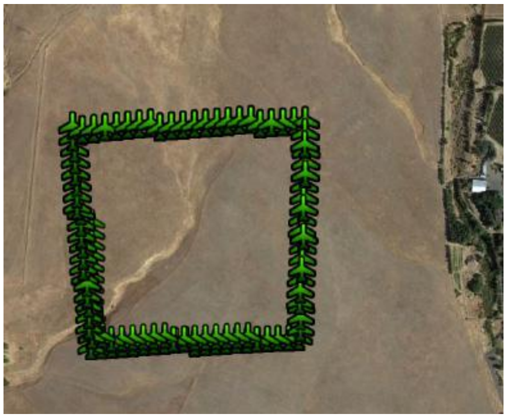


Figure 6.3: *SimpleFuser* simulated fused estimates.

ADS-B data stream was degraded but the LAMS data stream was not degraded. As a result of this, the *SimpleFuser* provided the LAMS estimates as the fused vehicle position estimates. This result is shown in Figure 6.3, where the green aircraft markers indicate the fused position estimates. From Figure 6.3 it can be seen that the fused estimates correspond to the LAMS estimated positions. The *SimpleFuser* allows the TRAPIS user to choose which estimate (ADS-B or LAMS) to use for the fused estimate, and therefore proved useful for scenarios in which one of the two data streams was degraded while the other was not.

6.1.3 KalmanFilterEstimator and WeightedFuser

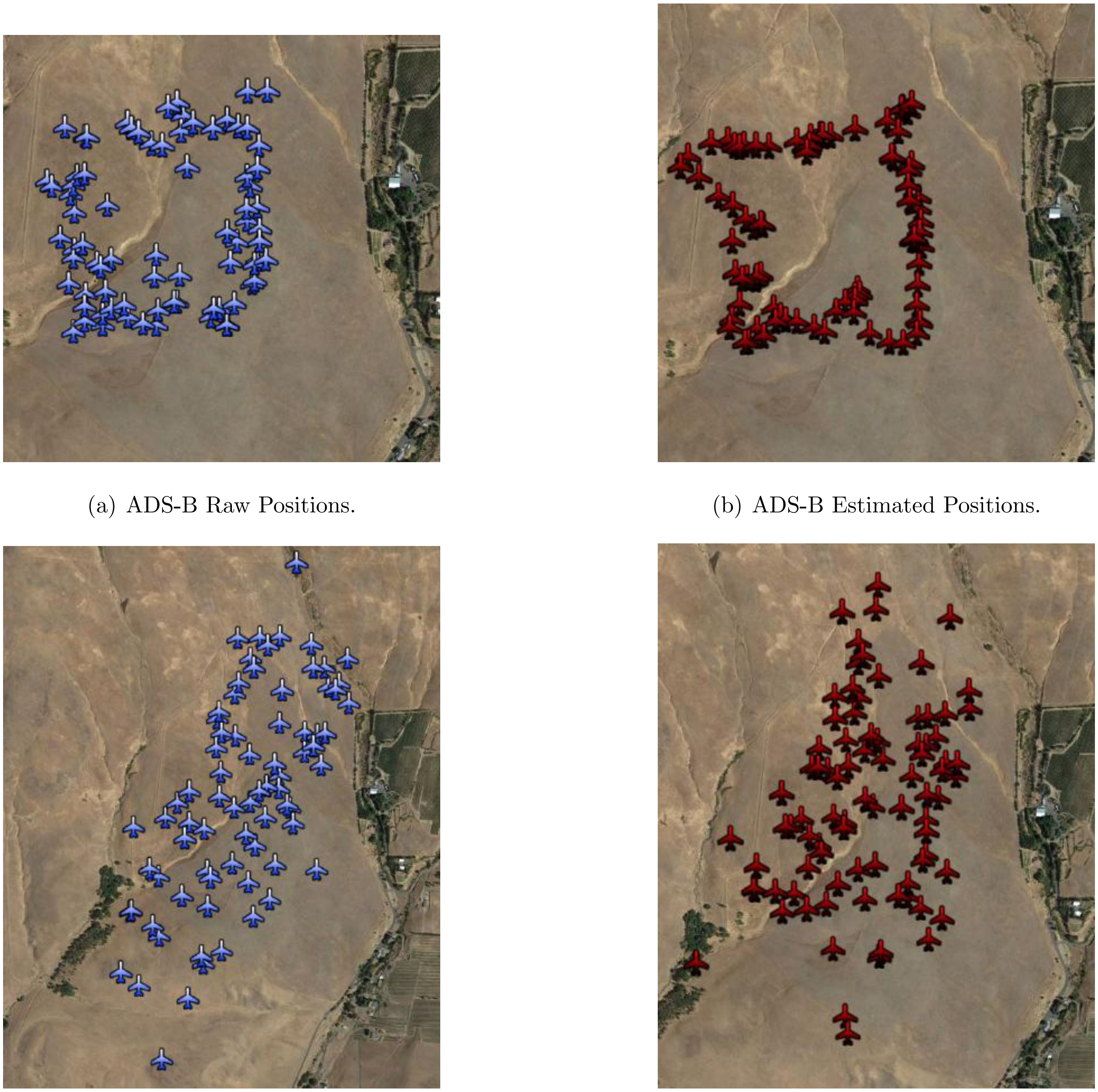
Although the *DoNothingEstimator* and the *SimpleFuser* proved desirable for providing accurate fused estimates in the presence of an unaltered data stream, these algorithms were not designed for use in the unlikely event that both data streams provide unreliable information to the TRAPIS software. In order to ensure that this case could be handled, the higher-level estimation and fusion algorithms were tested.

For the simulation in which the *KalmanFilterEstimator* was tested, the ADS-B data stream remained degraded to a NACp value of 8 for the entire simulation time span. The LAMS data stream was degraded to a NACp-equivalent of 6, thereby presenting position estimates to the TRAPIS software that were less-accurate than the ADS-B estimates. Although this scenario presented an unlikely case in which both data streams were degraded, the degradation of both data streams allowed the *KalmanFilterEstimator* to be used effectively, and presented the *WeightedFuser* with disparate error values so that the ADS-B and LAMS estimates could be effectively fused into a final, single estimate.

After the simulation was conducted, the raw and estimated positions from both data streams were compared to one another as shown in Figure 6.4. Based on the images shown in the figure, it can be seen that the *KalmanFilterEstimator* successfully reduced the error associated with the raw data streams to create more-accurate estimates of the sUAS position. Since the *KalmanFilterEstimator* was designed to assume a standard data stream error equivalent to a NACp value of 9, the estimates generated for the ADS-B data stream proved to be more accurate than the estimates generated for the LAMS data stream. This result was expected based on the design and implementation of the *KalmanFilterEstimator* algorithm.

Using the estimates generated for the ADS-B and LAMS data streams, the *WeightedFuser* weighted the position estimates associated with each data stream based on the true error of each stream in order to fuse the estimates. The fused estimates generated for the sUAS during a single orbit of the simulated flight path are shown in Figure 6.5. Based on the position estimates shown in the figure, it can be seen that the *WeightedFuser* accurately weighted the contributions of the ADS-B and LAMS estimates to provide a consistent, fused estimate of the aircraft position. Since the error associated with the ADS-B data stream was less than the error associated with the LAMS data stream for this simulation, the fused estimates generated by the *WeightedFuser* more closely resembled the ADS-B position estimates than the LAMS estimates.

From the results of the second simulation, it was shown that the *KalmanFilterEstimator* provided reasonably-accurate estimates of true aircraft positions in the presence of signal degradation. Since the *KalmanFilterEstimator* assumed a standard error corresponding to a NACp value of 9, the estimation algorithm more-accurately filtered the error out of the raw ADS-B data stream than the raw LAMS data stream. In addition to the estimation algorithm, the *WeightedFuser* data fusion algorithm provided a consistent, reasonable esti-



(a) ADS-B Raw Positions.

(b) ADS-B Estimated Positions.

(c) LAMS Raw Positions.

(d) LAMS Estimated Positions.

Figure 6.4: *KalmanFilterEstimator* simulated ADS-B and LAMS raw and estimated positions.

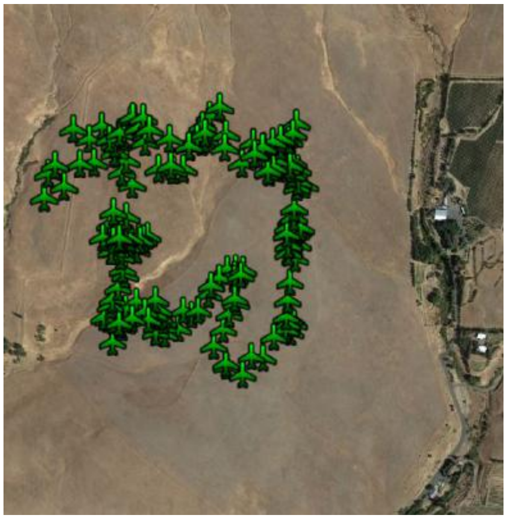


Figure 6.5: *WeightedFuser* simulated fused estimates.

mate of the sUAS position that represented the weighted average of the ADS-B and LAMS position estimates.

6.1.4 DynamicKalmanFilterEstimator and KalmanFuser

Once the initial estimation and fusion algorithms were tested through simulation, the final pair was tested. Using the same simulation scenario presented in the previous section, the *DynamicKalmanFilterEstimator* estimation algorithm and the *KalmanFuser* fusion algorithm were tested. These two algorithms were the final algorithms used for the TRAPIS software during the culminating flight demonstration, and thus extensive simulation testing was performed using these algorithms before flight tests began.

The raw and estimated positions from the ADS-B and LAMS data streams are shown in Figure 6.6. From the results presented in the figure, it can be seen that the *DynamicKalmanFilterEstimator* properly filtered the ADS-B positions and generated an accurate estimate of the rectangular sUAS flight path. Furthermore, it can be seen that the *DynamicKalmanFilterEstimator* generated a more accurate estimate from the provided LAMS positions than the *KalmanFilterEstimator*. Although error remained in the estimated LAMS positions, the magnitude of the error was significantly reduced, and the rectangular flight path can begin to be resolved from the filtered data stream.

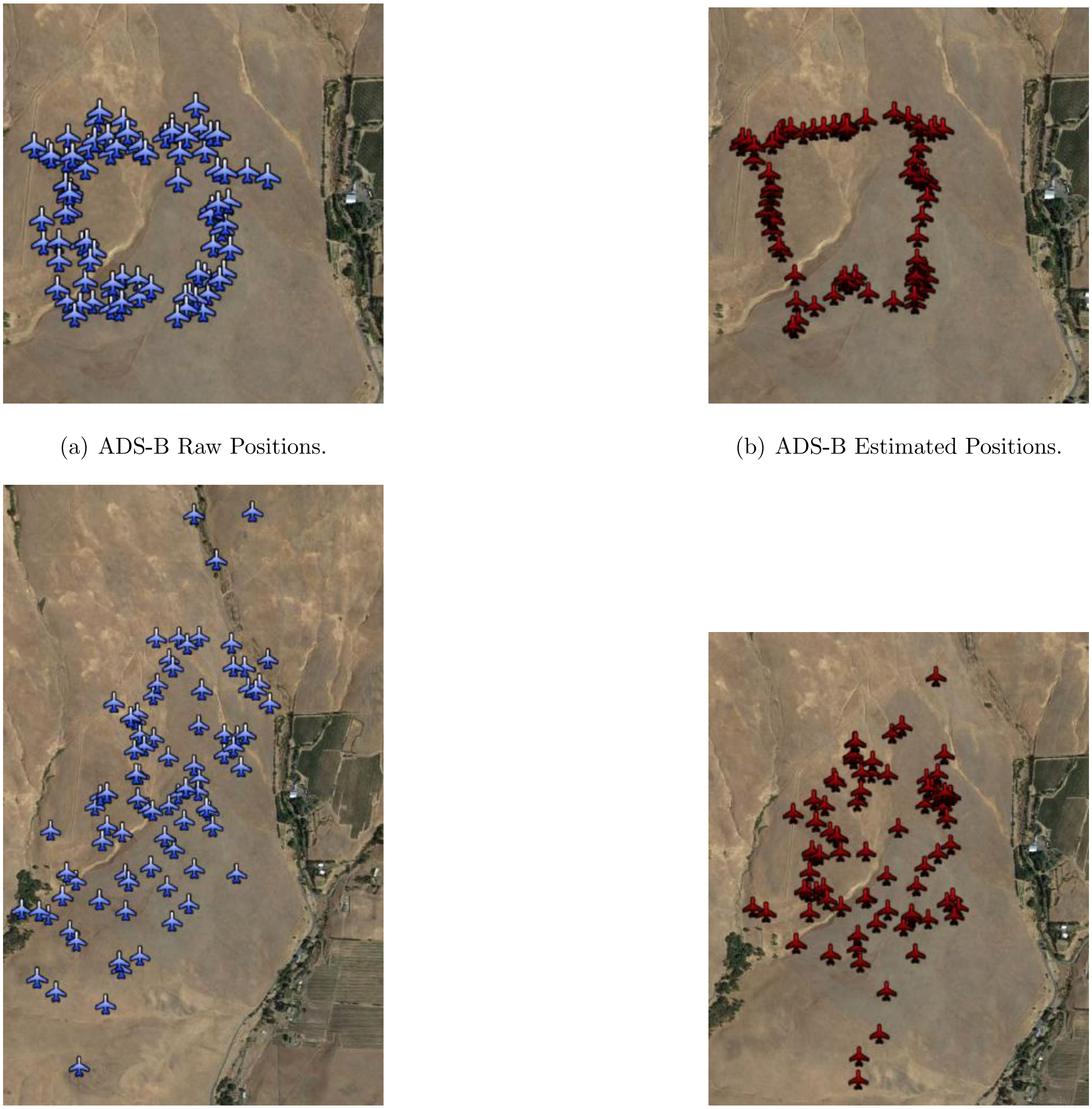
Once the position estimates were generated for the ADS-B and LAMS data streams, the *KalmanFuser* provided fused estimates to the TRAPIS software. Figure 6.7 shows the fused estimates from the simulation. The fused data stream closely resembles the filtered ADS-B estimated data stream shown in Figure 6.6(b), a result that was expected based on the weighting associated with the *KalmanFuser* algorithm. During simulations in which the ADS-B and LAMS data streams were degraded, the *DynamicKalmanFilterEstimator* provided the most accurate estimates of sUAS position, and the *KalmanFuser* provided the most accurate fused position estimates. This result was expected, and validated the decision to use the *DynamicKalmanFilterEstimator* and the *KalmanFuser* in the final iteration of the TRAPIS software.

6.1.5 GPS-Denied Operation

The final simulation run was designed in order to test the functionality of the estimation and fusion algorithms during GPS-denied operations. For purposes of simulating GPS denial, TRAPIS packets associated with the ADS-B data stream were not sent to the TRAPIS software during a period of 60 seconds from a simulation time of 40 seconds to 100 seconds. The *DynamicKalmanFilterEstimator* and *KalmanFuser* were used for the estimation and fusion, respectively, and the total simulation time remained constant at 350 seconds.

The estimated positions from the ADS-B and LAMS data streams are shown in Figure 6.8. Both of the images within Figure 6.8 show the position data associated with the first full lap of the flight path. It can be seen that the ADS-B data stream was only available for the first quarter-lap of the flight path before the GPS was artificially denied. Although the ADS-B data stream did not continue after this point, the LAMS data stream continued for the full lap of the flight path.

As the position estimates were being generated for the GPS-denied simulation, the *KalmanFuser* data fusion algorithm generated the fused position estimates as shown in



(a) ADS-B Raw Positions.

(b) ADS-B Estimated Positions.

(c) LAMS Raw Positions.

(d) LAMS Estimated Positions.

Figure 6.6: *DynamicKalmanFilterEstimator* simulated ADS-B and LAMS raw and estimated positions.

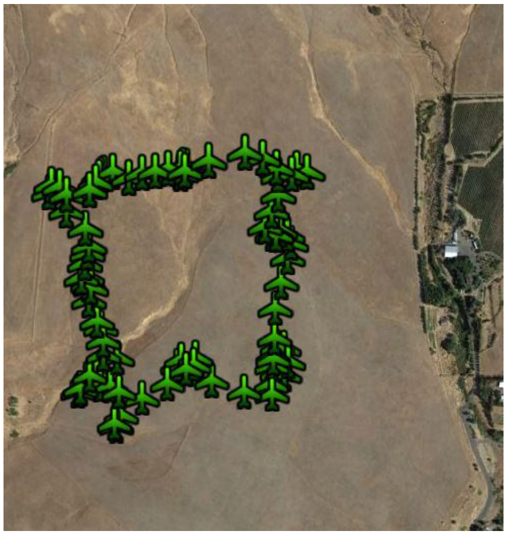
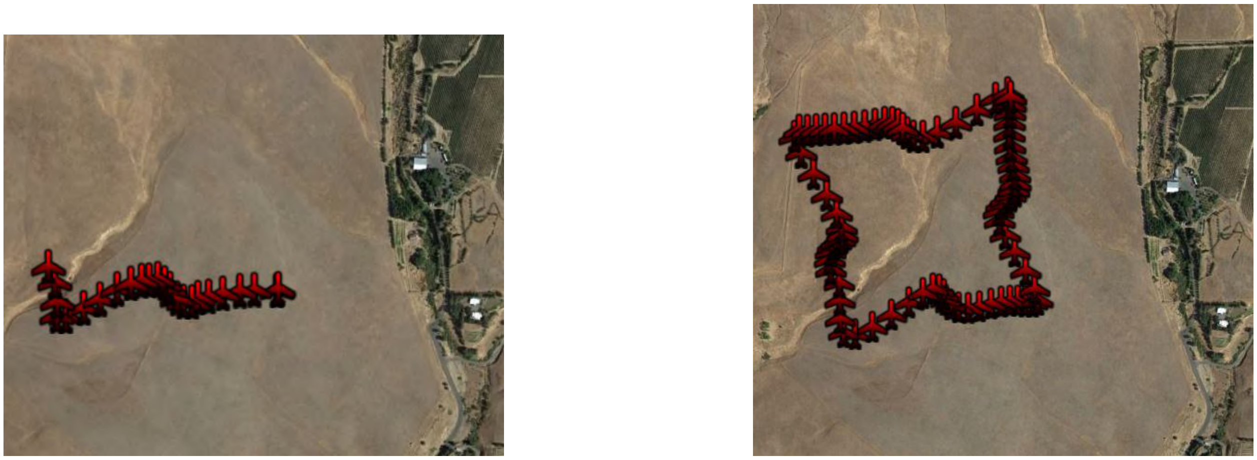


Figure 6.7: *KalmanFuser* simulated fused estimates.



(a) ADS-B Estimated Positions.

(b) LAMS Estimated Positions.

Figure 6.8: GPS-Denied simulated ADS-B and LAMS estimated positions.

Figure 6.9. The fused positions shown in Figure 6.9 represent the fused position estimates for the first one and a half laps of the flight path. From the figure, it can be seen that fused position estimates were present for the entire first lap of the flight path, despite the loss of GPS signal. Once GPS signal was regained and the ADS-B data stream continued, the fused

estimates seamlessly transitioned back to incorporating the position estimates from both the ADS-B and LAMS data streams.

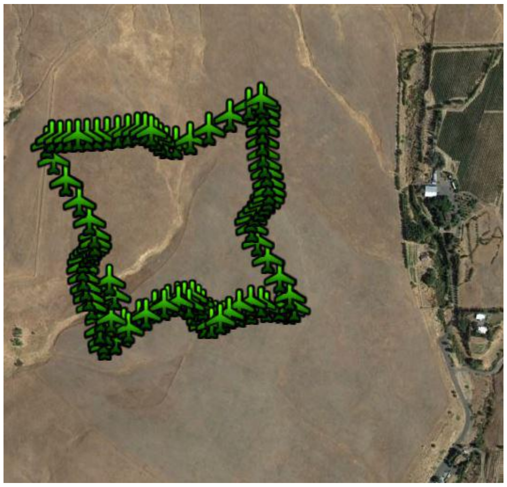


Figure 6.9: GPS-Denied simulated fused estimates.

The results of all simulations associated with the TRAPIS software showed that all of the estimation and fusion algorithms performed as expected and provided reasonable position estimates. Most importantly, the simulations showed that the data fusion algorithms provided a reasonable fused position estimate stream to the TRAPIS software regardless of the availability or accuracy of the ADS-B and LAMS data streams.

6.2 Meadowbrook Test Results

After the TRAPIS code was extensively tested and verified in simulation, preliminary flight tests were conducted at Meadowbrook Farms in North Bend, WA. The purpose of these flight tests were to ensure that the TRAPIS payload and associated software worked to track the actual sUAS in real time. These flight tests were performed during three separate excursions on August 25th, September 9th, and September 16th, 2016.

6.2.1 Initial Flight Testing

In order to test the altitude reporting capabilities of the ADS-B transponder while testing the full capabilities of the sUAS autonomous flight path tracking, the altitudes were varied on alternating legs of the rectangular flight path. While transitioning between the first and second waypoints, the aircraft climbed from an altitude of 250 feet AGL to an altitude of 350 feet AGL. The aircraft remained at 350 feet AGL between the second and third waypoints before descending back to 250 feet AGL between the third and fourth waypoints. After reaching the fourth waypoint, the aircraft remained at 250 feet AGL while returning to the first waypoint. During the testing, a total of 5 flight path circuits were flown. Once the final circuit was completed, the aircraft returned to the ground station home location and began orbiting at 250 feet AGL before the remote pilot took over manual control of the aircraft for landing. Initial testing showed that the aircraft was able to autonomously follow the designated flight path while climbing and descending as necessary to reach the appropriate altitudes.

After the aircraft demonstrated the ability to autonomously follow the designated flight path without the transponder operating, the transponder was set to altitude reporting mode and the flight path was followed again. During the flight test, the ADS-B transponder associated with the TRAPIS payload was seen with both the AFSL TRAPIS software and the iPad WingX Pro application through the use of the Sagetech Clarity ADS-B In receiver. During the first test, the GPS integrity was not artificially degraded, and the aircraft performed as expected. The ADS-B tracking output from the flight test can be seen in Figure 6.10 and Figure 6.11, both of which are oriented with North at the top of the image.

Based on these figures, it can be seen that the ADS-B information corresponded to the true flight path flown by the sUAS, and that the altitude reporting capability of the sUAS matched the desired altitudes for the designated flight path. During testing there was a 6 to 9 knot variable wind from the North at the sUAS flight altitudes which led to the flight path overshoots seen between the second, third, and fourth waypoints shown in Figure

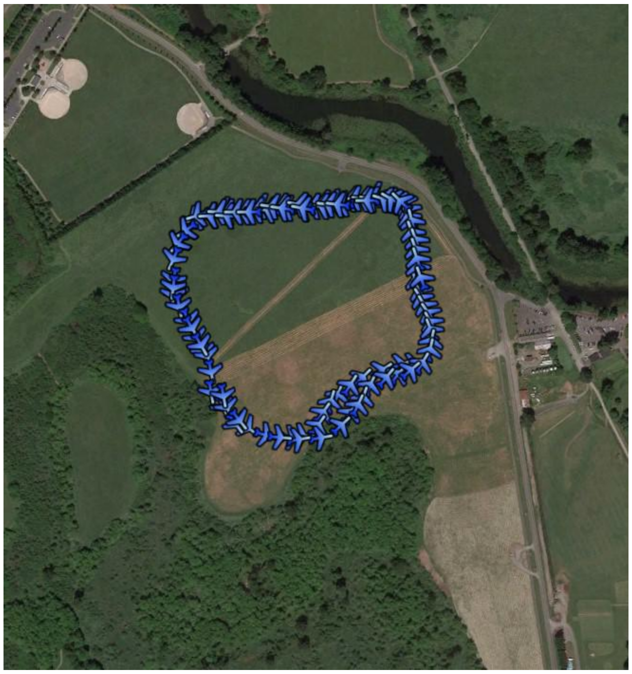


Figure 6.10: Initial test ADS-B position tracking results.

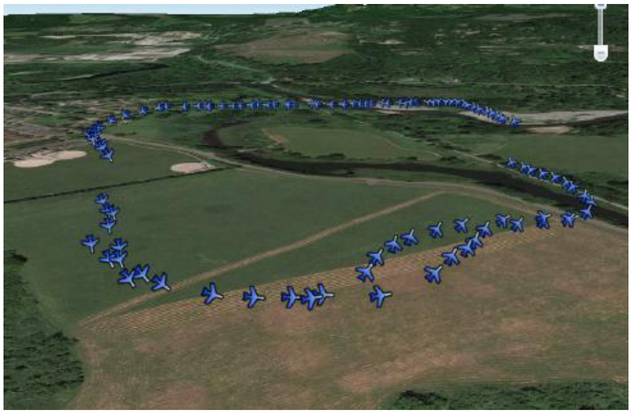


Figure 6.11: Initial test ADS-B altitude tracking results.

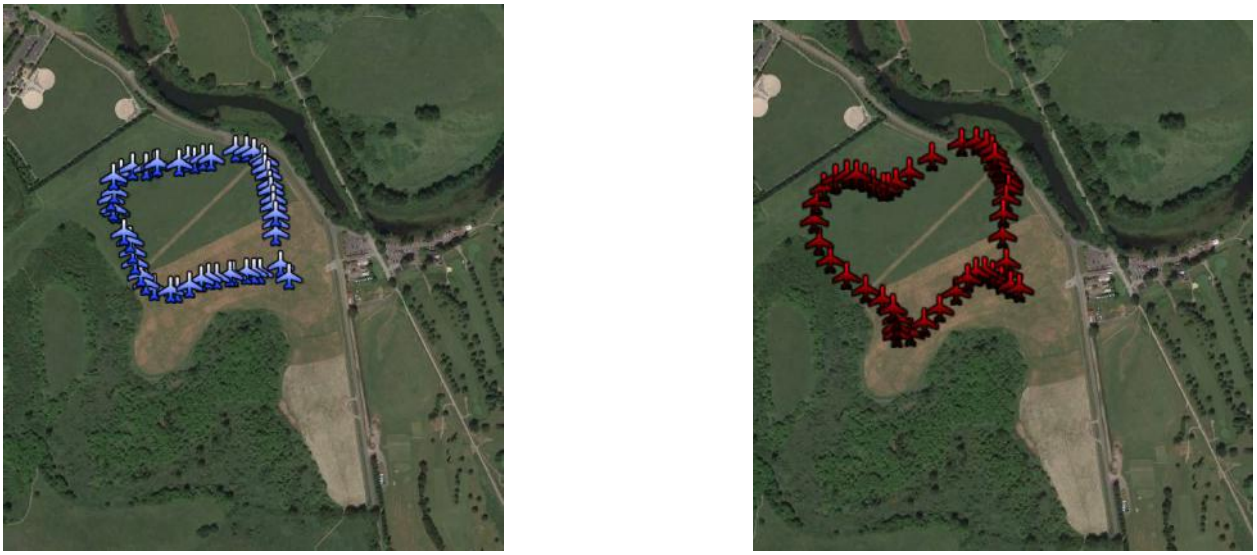
6.10. Additionally, transponder pressure altitude resolution is ±100 feet, which accounted for the two distinct altitudes shown in Figure 6.11. After the initial tests were completed, the aircraft was flown during two additional tests on September 9th and September 16, 2016. The tests completed on these later two tests dates were designed to mirror the tests designed for the final flight demonstration in the airspace around the KDLS airport. Additional results associated with these flight tests are detailed in [21]. For the analysis contained in this thesis, the performance of the position estimation algorithms will be considered. Due to software restrictions, the test team was unable to accurately simulate LAMS tracks at the Meadowbrook Farms flight test location during flight testing. Since simulated LAMS data was being generated for a circular flight path flown around the KDLS airfield during the sUAS flight testing at Meadowbrook Farms, the fused position estimates were located in southern Washington and did not provide meaningful or accurate data. As a result, the data fusion algorithms were verified during the culminating flight test at the KDLS airfield, and only ADS-B estimation results were analyzed for the Meadowbrook Farms testing.

During all testing of the sUAS with the ADS-B transponder at Meadowbrook Farms the final iteration of TRAPIS software was used, to include the *DynamicKalmanFilterEstimator* estimation algorithm and the *KalmanFuser* fusion algorithm. The software used for the flight testing at Meadowbrook Farms matched the TRAPIS software used for the final flight demonstration at the KDLS airfield.

6.2.2 Normal GPS Operation Testing

During the Meadowbrook Farms flight testing, all three modes associated with the ADS-B transponder payload were tested, namely the non-degraded GPS, degraded GPS, and denied GPS modes. Initially the transponder payload was tested without GPS degradation in order to ensure that the estimation algorithms worked as desired while receiving actual ADS-B information. It was found that the ADS-B information was received at the ground station regardless of the aircraft location on the flight path and associated transponder antenna orientation. The raw and estimated position results of the testing with normal GPS operation are shown in Figure 6.12.

From the figure, it can be seen that during normal operation, the 3DR GPS unit provided accurate position information to the ADS-B transponder. The ADS-B signal was not interrupted or lost during the course of testing, and the positions provided to the TRAPIS



(a) ADS-B Raw Positions.

(b) ADS-B Estimated Positions.

Figure 6.12: Initial flight test raw and estimated positions with normal GPS operation.

software represented the true aircraft positions. Additionally it can be seen that the estimated positions closely aligned with the true aircraft positions, although there was error associated with the estimates. This error was expected due to the nature of the discrete Kalman filter estimation algorithm. When the aircraft made turns at the waypoints, the matrices associated with the Kalman filter had to update to account for the change in the velocity vector of the aircraft. As a result of the fact that several filter update cycles were required for the matrices to adjust, overshoots were seen in the estimated positions.

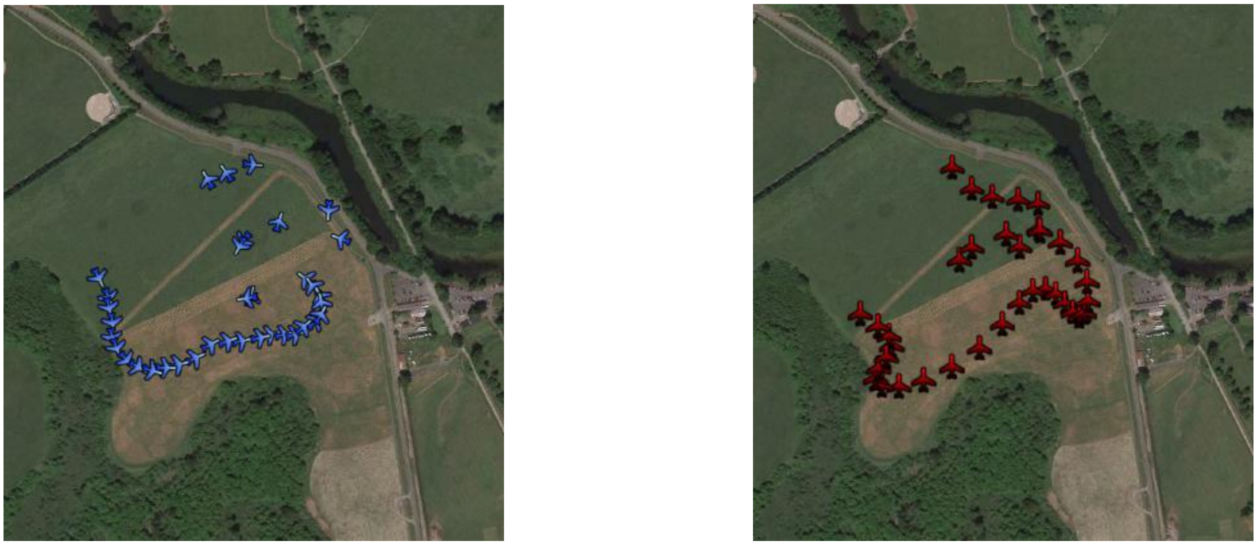
The testing with normal GPS operation detailed both the accuracy of the ADS-B information during normal operation, and the suitability of the consumer-level 3DR GPS unit for use with the ADS-B transponder payload. It was demonstrated that in ideal situations where the GPS signal is not degraded or denied, the raw ADS-B data stream results should be used as the position estimates. Nonetheless, the position estimates provided by the *DynamicKalmanFilterEstimator* closely matched the true aircraft positions, especially when considering the relatively small maneuvering space and quick velocity vector changes of the sUAS aircraft.

6.2.3 GPS-Degraded Operation Testing

After the estimation algorithm was tested with normal GPS operation, the algorithm was tested during GPS-degraded operations. Based on the Arduino code associated with the transponder payload, the artificial GPS degradation added noise to the GPS information provided to the ADS-B transponder commensurate with a NACp value of 8 [16]. Along with position information, the velocity information associated with the GPS data being provided to the ADS-B transponder was degraded. Unfortunately, this degradation caused problems for the estimation algorithms since the GPS velocities were no longer directly correlated with the GPS position information. Nonetheless, the estimation algorithm provided improved estimates of true sUAS position when compared to the raw ADS-B data stream information.

The raw and estimated data streams for a portion of one flight path lap flown with GPS-degradation are shown in Figure 6.13. From the figure, it can be seen that the GPS-degradation began once the aircraft turned to the North. Between waypoint 1 and waypoint 2 the GPS signal was artificially degraded, and the aircraft positions from the ADS-B data stream no longer represented the true aircraft positions. By comparison, it was shown that although the estimated aircraft positions included error, the estimated positions were more closely aligned with the true flight path than the raw positions.

Although the estimated positions proved to match the true aircraft flight path more closely than the raw positions during GPS-degraded operations, errors associated with the estimated positions remained. These errors were likely associated with the uncorrelated nature of the degraded positions and degraded velocities. Nonetheless, the flight test showed that the *DynamicKalmanFilterEstimator* provided reasonably accurate estimates of sUAS position in the presence of GPS degradation, and these estimates more closely matched the true aircraft positions than the raw positions.



(a) ADS-B Raw Positions.

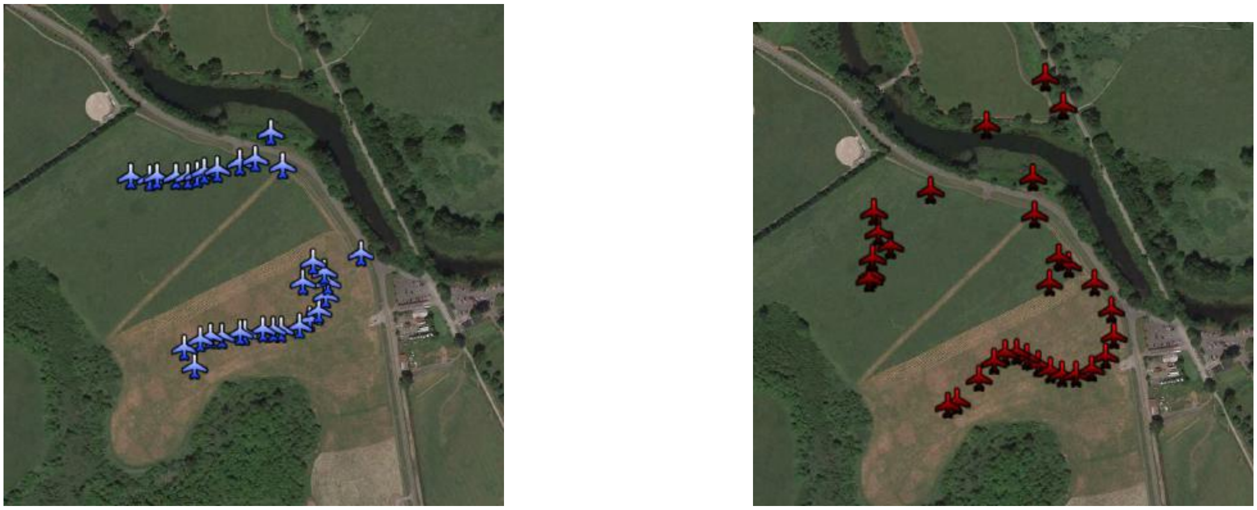
(b) ADS-B Estimated Positions.

Figure 6.13: Initial flight test raw and estimated positions with degraded GPS operation.

6.2.4 GPS-Denied Operation Testing

After the normal and degraded GPS modes were tested, the GPS-denied payload mode was tested. Since estimates are not generated when the GPS signal is denied, the test was designed to ensure that the estimation algorithm would continue to provide accurate estimates of aircraft position once the GPS signal was no longer being denied. The raw and estimated position results of the test are shown in Figure 6.14.

From the images in 6.14, it can be seen that the GPS signal was denied between waypoint 1 and waypoint 2. The GPS signal returned after the aircraft turned from waypoint 2 and began flying towards waypoint 3. Upon initial inspection, it can be seen that the estimated positions do not accurately capture the track of the aircraft. As with the GPS-denied simulation results, the estimated positions show that once the GPS signal is restored after being denied, the Kalman filter requires several observations for the associated matrices to update and bring the estimates back towards the true aircraft track. Furthermore, in real scenarios the LAMS data stream would be present, and as a result the fused position estimates would



(a) ADS-B Raw Positions.

(b) ADS-B Estimated Positions.

Figure 6.14: Initial flight test raw and estimated positions with denied GPS operation.

more closely match the true aircraft track than the ADS-B position estimates. Although the position estimates provided by the *DynamicKalmanFilterEstimator* during GPS-denied operations were not as accurate as those generated during normal and GPS-degraded operations, the test showed that the estimation algorithm would continue to provide position estimates once the GPS signal was restored after temporary denial.

6.3 Simulation Comparison Data

From the simulation and initial flight testing results, the estimated and fused position information showed reasonable agreement with true aircraft positions. From the simulation results, it was found that there was a maximum position error of approximately 100 feet for the simulated sUAS aircraft transiting the rectangular flight path with degraded GPS and LAMS information using the *KalmanFilterEstimator* estimation algorithm and the *WeightedFuser* fusion algorithm. This maximum error between the estimated and true positions corresponded to a 10% error when compared to the total flight path length of approximately 1000 feet.

Although no fused estimates were generated for the initial flight testing, the estimated position results showed similar error when compared with the simulation results. The maximum position error was approximately 100 feet which accounted for a 10% error when compared to the square flight path side length of approximately 1000 feet.

Chapter 7

FLIGHT DEMONSTRATION RESULTS

7.1 ADS-B and LAMS Raw Data

7.1.1 Manned Aircraft Data

During flight testing, unforeseen environmental factors and issues with the sUAS ADS-B transponder prevented sUAS data from being gathered simultaneously with manned aircraft data as originally planned. These challenges are summarized in Section 7.4 of this report. As a result of these issues, sUAS ADS-B and LAMS data was gathered on the second day of testing, while manned aircraft data was gathered on the first day of testing. Despite the fact that data was not gathered simultaneously for the manned aircraft and the sUAS, both data sets include all desired information from the test plan. In addition to the difficulties faced in gathering the manned aircraft and sUAS data simultaneously, only one of the manned aircraft associated with the flight demonstration was tracked during testing. Since the Vans RV-12 was fitted with an ADS-B Out transponder, it was tracked using both the ADS-B In receiver and the LAMS station. The Cessna 172 did not have an ADS-B Out transponder, but it had a traditional Mode C transponder. Based on this difference in equipment between the two aircraft, it was planned that the Cessna 172 would be tracked using only the LAMS station, which is capable of tracking aircraft with traditional Mode C transponders. During testing, it was found that the LAMS station did not track the Cessna 172, and as a result no data was gathered for the aircraft. It is speculated that the Cessna 172 transponder might have suffered from the same issues as the original sUAS transponder unit detailed in Section 7.4 of this report. Transponder pulse widths outside of specified ranges could have contributed to the lack of LAMS tracking for the Cessna 172, but this hypothesis has not been verified.

The RV-12 pilot was tasked with flying the aircraft around the flight path to the southeast of the KDLS airport as shown in Figure 4.8. This flight path involved flying counterclockwise patterns with altitudes varying between 3,500 feet MSL and 4,500 feet MSL. The pilot remained on station with the aircraft for approximately one hour, and during that time flew continuous circuits of the designated flight path. Data gathered for the RV-12 during testing to include ADS-B, LAMS, and secondary GPS data corresponding to one lap of the flight path is shown in Figure 7.1.

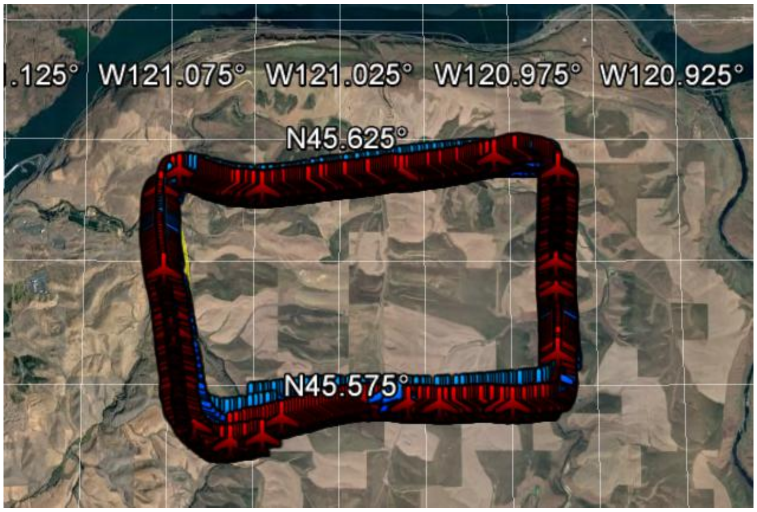
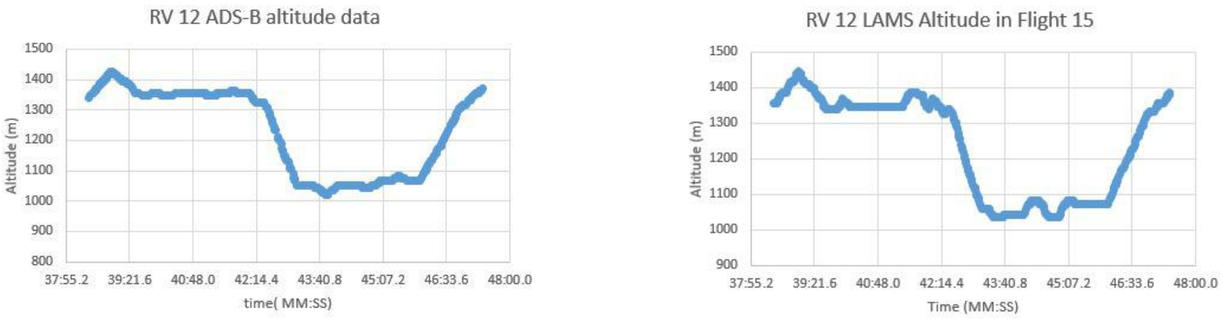


Figure 7.1: RV-12 HiL unit data flash logs (yellow line), ADS-B (blue aircraft) and LAMS position data (red aircraft).

The data in Figure 7.1 shows all sources of aircraft positioning information for the RV-12 plotted simultaneously. The figure shows the ADS-B data represented as blue aircraft, the LAMS data represented as red aircraft, and the secondary GPS data from the hardware-in-the-loop unit is represented as yellow lines. The information plotted in the figure corresponds to the position information gathered for one lap of the pre-designated flight path. During testing, 30 to 40 mph winds out of the north were experienced. The effect of the high winds can be seen in the figure by the presence of flight path overshoots to the south when the aircraft turned from a southbound heading to an eastbound heading. Additionally, the rectangular flight path is visibly slanted due to the effect of the high winds on the overall ground track of the aircraft as it transited between the flight path coordinates.

The tracks presented in Figure 7.1 show strong agreement between the ADS-B position information and the LAMS position information, with a maximum difference of approximately 0.35 nautical miles at the southwest corner of the flight path. Additionally, the aircraft ADS-B track shown by the small blue aircraft symbols closely match the GPS tracks generated by the secondary hardware-in-the-loop GPS unit shown by the yellow lines.

In addition to comparison of the position information provided by the ADS-B and LAMS tracks, altitude information associated with the tracks was compared. The altitude information recorded with the ADS-B data was gathered from the on-board GPS unit associated with the RV-12 aircraft and represented a geometric altitude. The altitude information recorded with the LAMS data was gathered from the pressure altimeter associated with the transponder, and represented a pressure altitude. The time histories of this altitude information were plotted for one circuit of the flight path. The geometric altitude time history is shown in Figure 7.2(a), and the pressure altitude time history is shown in Figure 7.2(b). It is important to note that the pressure altitudes reported by the aircraft transponder have been corrected to true altitudes by accounting for nonstandard pressure at the LAMS station during testing. The pressure-corrected, true altitude data is the data shown in Figure 7.2(b).



(a) RV-12 ADS-B altitude data.

(b) RV-12 LAMS altitude data.

Figure 7.2: RV-12 ADS-B and LAMS altitude data.

After comparing the data presented in Figure 7.2(a) and Figure 7.2(b), it can be seen that the reported altitudes closely match. In both figures, the altitude differences between flight path legs are clearly shown, with climbs and descents represented as well. Initially, the aircraft maintained an altitude of 1350 meters before descending to an altitude of 1050 meters. These altitudes correspond to the desired flight path altitudes of 4,500 feet MSL and 3,500 feet MSL for the westbound and eastbound legs of the flight path, respectively. The small altitude variations seen in 7.2(b) can be attributed to the altitude resolution of the transponder pressure altitude encoder. Transponders encode pressure altitudes at 25 foot increments while GPS altitude measurements have higher resolutions. This equipment difference accounted for the small differences seen between the plots shown in Figure 7.2(a) and Figure 7.2(b).

7.1.2 sUAS Normal GPS Data

All ADS-B and LAMS data for the sUAS aircraft was gathered on the second day of testing after the flight team ground station location had been moved to the secondary location. Two successful flights were conducted with the sUAS aircraft on the second day of testing, and all of the desired GPS scenarios were tested. The most complete data set was gathered during the second flight of the sUAS aircraft, after the ADS-B antenna was changed to a vertical orientation.

The ground tracks from a single lap segment of the ADS-B and LAMS data associated with the second successful sUAS flight are shown in Figure 7.3. In the figure, it can be seen that the ADS-B data shown by the blue aircraft symbols does not directly align with the LAMS data shown by the red aircraft symbols. While this difference seems to indicate a large discrepancy between the ADS-B and LAMS position information, it is important to consider the scale of the sUAS flight path in relation to the manned aircraft flight path. The manned aircraft flight path was flown with side lengths of 4.5 and 2.7 NM for the long and short sides, respectively, and represented an area of 39 square nautical miles. By comparison, the altered sUAS flight path involved approximately rectangular laps with long side lengths of 0.15 NM and short side lengths of 0.1 NM, accounting for a total area of 0.015 square nautical miles. Additionally, the maximum error between the ADS-B track and LAMS track for the manned aircraft was 0.35 NM, a distance over two times as large as the longest side length for the sUAS rectangular path. By comparison, the maximum error between the ADS-B ground track and the LAMS ground track for the sUAS test flight was 0.1 NM.

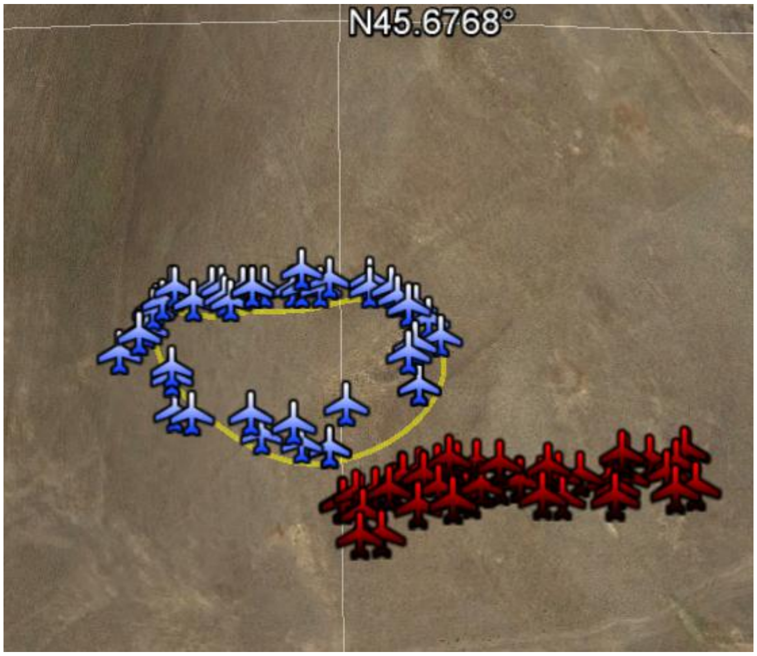


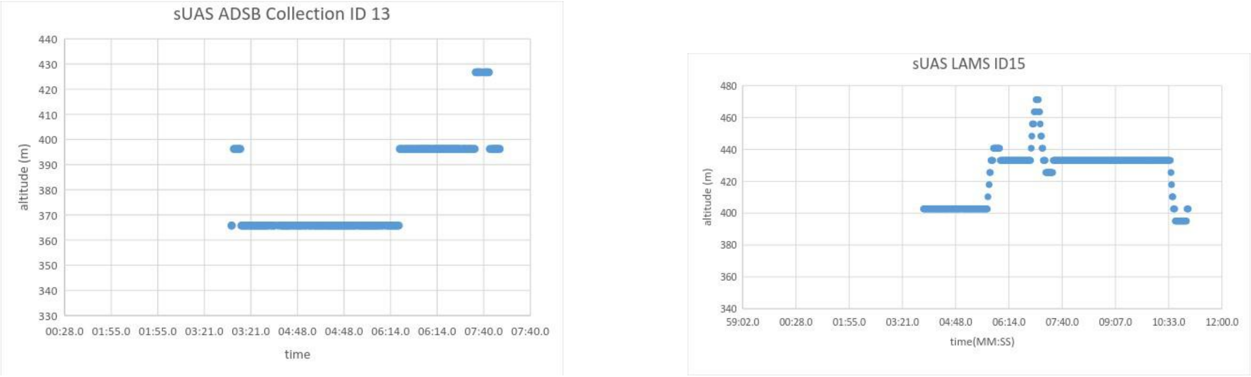
Figure 7.3: sUAS HiL unit data flash logs (yellow line), ADS-B (blue aircraft) and LAMS position data (red aircraft).

Once position information from the ADS-B and LAMS sources had been compared, the altitude information was compared. On the manned aircraft, ADS-B altitudes were reported as geometric altitudes gathered from the on-board GPS unit, and LAMS altitudes were reported as pressure altitudes from the transponder unit and corrected to true altitudes at the LAMS station. On the sUAS aircraft, these altitudes were gathered and reported in the same manner, however the GPS and transponder equipment used was much different. The GPS used on the sUAS for ADS-B altitude reporting was a non-WAAS 3DR uBlox GPS unit. Additionally, the pressure altitude reported to the LAMS station was gathered directly from the Sagetech XPS-TR transponder. Since investigation and placement of a static pressure port on the external surface of the sUAS airframe was outside the scope of this research, the static pressure required for the pressure altitude was gathered by the transponder pressure altimeter directly inside the payload bay.

The altitudes reported in the sUAS ADS-B packets are shown in Figure 7.4(a) for a four minute period of the test flight. Since all flight operations had to be conducted below 400 feet AGL, the range of allowable altitudes for the sUAS was much lower than the range of allowable altitudes for the manned aircraft. This fact coupled with the fact that geometric altitude variations were only resolved at 30 to 40 meter increments with the Sagetech transponder resulted in the discontinuities seen in the figure.

The altitudes reported in the sUAS LAMS packets are shown in Figure 7.4(b) for a twelve minute period of the test flight which covers the four minute period referenced in Figure 7.4(a). As with the ADS-B altitudes, the LAMS altitudes plotted in Figure 7.4(b) show significant discontinuities. Just as with the manned aircraft LAMS altitude data, these discontinuities can be attributed to the resolution of the transponder pressure altitude encoder. Since the sUAS flights occurred in a much narrower altitude band, the discontinuities were much more pronounced than they were for the manned aircraft data. The XPS-TR transponder is programmed to output altitudes in 100 foot increments, and the discontinuities of 30 to 40 meters seen in the plots coincide with this published 100 foot altitude resolution.

By initial comparison, the altitude information presented in the two figures seems to differ significantly during the matched periods on the time scales, on the order of 40 meters. When compared to the manned aircraft altitude data from the ADS-B and LAMS sources, it does appear that the sUAS altitude data was much less accurate. Several factors could have contributed to this discrepancy to include placement and calibration of the transponder pressure altimeter, and altitude resolution capabilities of the 3DR uBlox GPS unit. Since no external static pressure source was placed on the sUAS airframe, the static pressure referenced for the transponder-based pressure altimeter was the static pressure inside the payload bay. This



(a) sUAS ADS-B altitude data.

(b) sUAS LAMS altitude data.

Figure 7.4: sUAS ADS-B and LAMS altitude data.

reference pressure could have contributed to the higher reported altitudes seen in the LAMS altitude data plot as compared to the ADS-B altitude data plot. The static pressure inside the payload bay would have been lower than a static pressure referenced from the outside of the aircraft, and as a result the pressure altitude seen by the transponder pressure altimeter would be higher inside the payload bay. In addition to the static reference pressure location inside the aircraft payload bay, the GPS used on the sUAS was a non-WAAS GPS which could have contributed to vertical position errors during flight testing. Ultimately, the altitude information presented in Figure 7.4 shows that the altitude measurements are reported at 100 foot intervals as expected from the transponder. The difference of approximately 40 meters between the pressure altitude data and the ADS-B altitude data can be attributed to a variety of potential factors, the most significant of which was the measurement of the pressure altitude inside the aircraft payload bay.

7.1.3 sUAS GPS-Degraded Data

After several rectangular laps were flown with the sUAS in manual mode and the GPS operating in a non-degraded state, the GPS information provided to the transponder was artificially degraded. By artificially degrading the GPS information being provided to the transponder the ADS-B position measurements were corrupted and the LAMS position measurements served as the primary means of tracking the aircraft. The results of the testing with the GPS information being artificially degraded are shown in Figure 7.5.

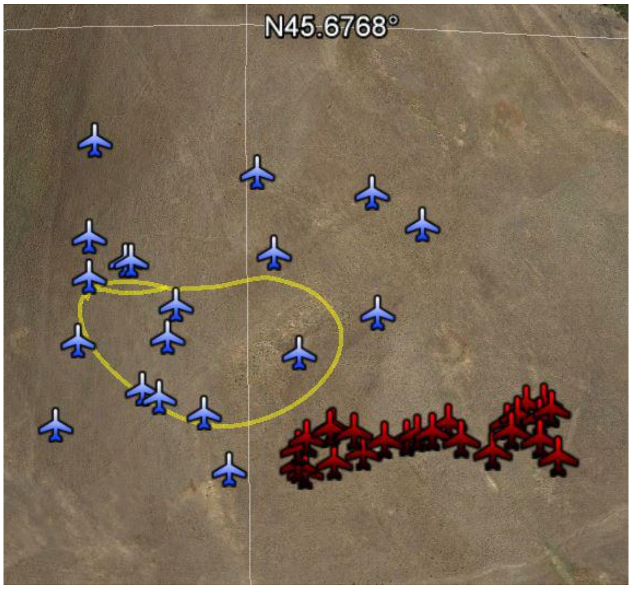


Figure 7.5: sUAS GPS-Degraded ADS-B (blue aircraft) and LAMS (red aircraft) position data.

Based on the aircraft position information shown in the figure, it can be seen that the GPS information provided to the transponder was artificially degraded. Based on the Arduino code, the positions, headings, and velocities associated with the GPS information were degraded in such a manner that the GPS information was unreliable for sUAS tracking on a small flight path. Although the degraded GPS information proved unreliable for tracking the sUAS during testing, the LAMS position information continued to be reliable, though slightly offset from the actual vehicle ground track. From the figure it can be seen that the LAMS position information continued to be received by the TRAPIS software during GPS-degraded operations, and the LAMS station sufficiently tracked the sUAS movements on the east-west flight path. The offset between the ADS-B track and the LAMS track was consistent with the offset for the tracks during normal GPS operations. Overall, the LAMS track showed that the LAMS could be reliably used as a secondary sUAS tracking system during GPS-degraded operations.

7.1.4 sUAS GPS-Denied Data

Once several additional laps were flown with artificial GPS degradation, the GPS signal provided to the ADS-B transponder was artificially denied. As with the GPS degradation, during GPS-denied operations the LAMS served as the primary means of sUAS tracking. The LAMS data associated with the GPS-denied testing is shown in Figure 7.6.

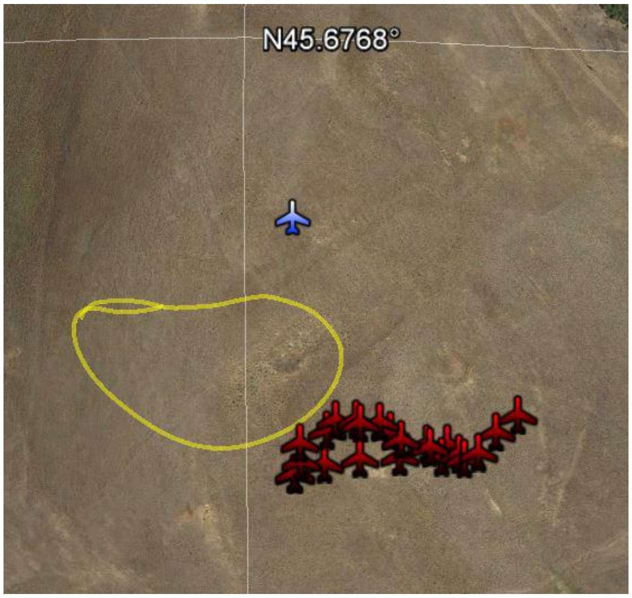


Figure 7.6: sUAS GPS-Denied ADS-B (blue aircraft) and LAMS (red aircraft) position data.

By looking at the figure, it can be seen that during GPS-denied operations the ADS-B information stopped being provided to the TRAPIS software. Once GPS information stopped being provided to the transponder, a single blue aircraft symbol remained to indicate the last reported GPS position. Although ADS-B information was not provided to the TRAPIS software during the GPS-denied operations, LAMS information continued to be provided. The LAMS track shown in the figure closely matches the LAMS track generated during

GPS-degraded operations. Based on the information gathered from the LAMS during GPS-denied operations, it was shown that the LAMS could be used as a reliable primary tracking method.

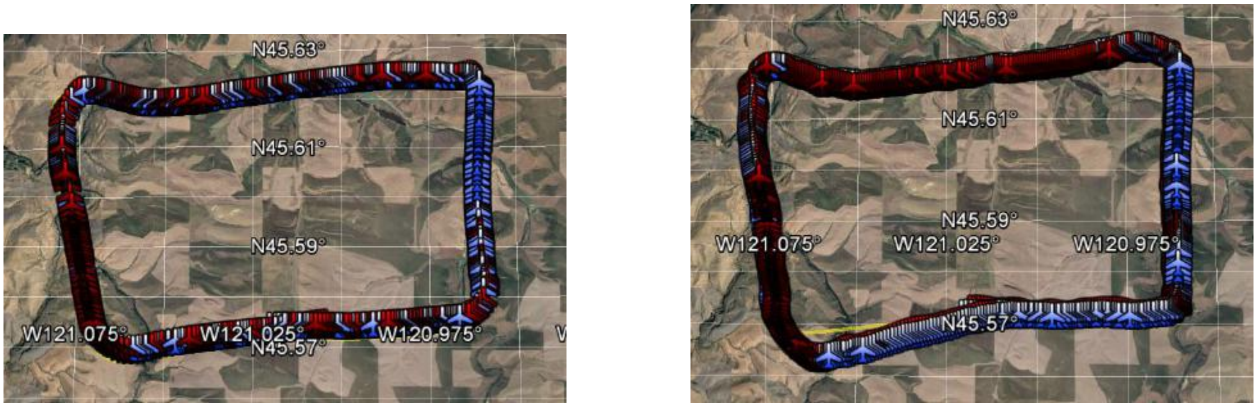
7.2 Estimated and Fused Data

7.2.1 RV-12 Estimation and Fusion

In the final iteration of the TRAPIS software, the *DynamicKalmanFilterEstimator* was used with the *KalmanFuser* to generate the position estimates and fused estimates, respectively. Since the ADS-B and LAMS information associated with the RV-12 was not degraded or denied, the position estimates closely resembled the raw position data. The ADS-B and LAMS estimated positions are shown with the raw positions in Figure 7.7. The estimated positions are shown by the red aircraft icons, while the raw positions are shown by the blue aircraft icons and the secondary GPS positions are shown by the continuous yellow lines. From the figure, it can be seen that the estimated positions closely match the raw positions for both the ADS-B and LAMS data streams. This result was expected since the data streams were not corrupted in any manner.

As the estimated ADS-B and LAMS positions were being generated, the fused estimate was simultaneously generated. Since both the estimated ADS-B and LAMS positions closely matched the raw positions, the fused estimate closely matched the raw positions as well. The fused position estimates are shown in Figure 7.8. The estimated positions are shown by the green aircraft icons, while the secondary GPS positions are shown by the continuous yellow line. From the data presented in the figure, it can be seen that the fused position estimates closely matched the raw and estimated positions associated with the ADS-B and LAMS data streams.

Based on the results associated with the RV-12 for the final flight demonstration, it was shown that the aircraft could be tracked using the TRAPIS software. Furthermore, the estimation and fusion results showed that the estimation and fusion algorithms provided



(a) RV-12 ADS-B raw (blue aircraft) and estimated (red aircraft) positions overlaid on data flash log track (yellow line).

(b) RV-12 LAMS raw (blue aircraft) and estimated (red aircraft) positions overlaid on data flash log track (yellow line).

Figure 7.7: RV-12 raw and estimated ADS-B and LAMS positions.

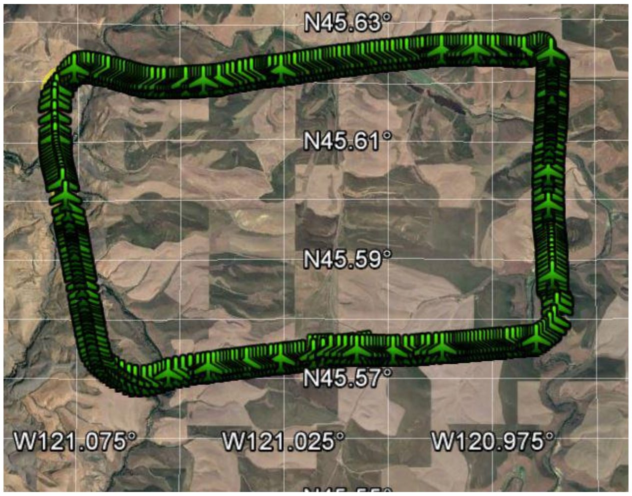
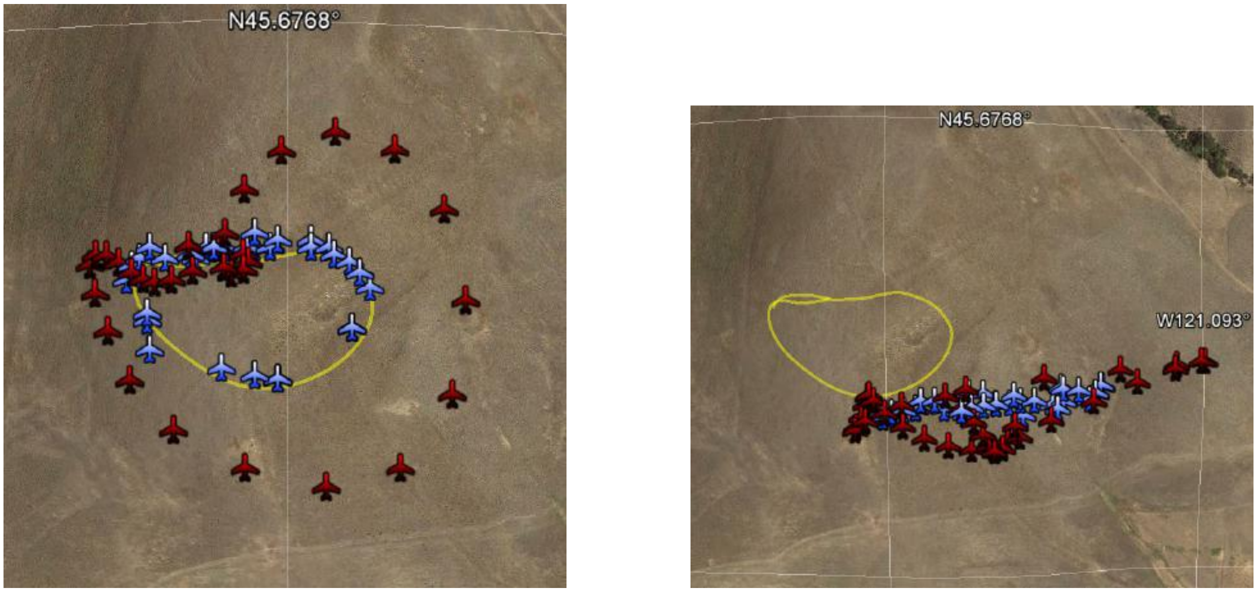


Figure 7.8: RV-12 fused (green aircraft) position estimates overlaid on data flash log track (yellow line).

accurate estimates of aircraft position based on the data associated with the ADS-B and LAMS data streams. Although the ADS-B and LAMS data streams were not degraded or denied, the results showed that the TRAPIS software performed as-desired during the final flight demonstration.

7.2.2 sUAS Estimation and Fusion

While the raw sUAS ADS-B and LAMS position information was provided to the TRAPIS software, the estimated positions were generated as shown in Figure 7.9. The estimated positions are shown by the red aircraft icons, while the raw positions are shown by the blue aircraft icons and the on-board GPS positions are shown by the continuous yellow lines.



(a) sUAS ADS-B raw (blue aircraft) and estimated (red aircraft) positions overlaid on data flash log track (yellow line).

(b) sUAS LAMS raw (blue aircraft) and estimated (red aircraft) positions overlaid on data flash log track (yellow line).

Figure 7.9: sUAS raw and estimated ADS-B and LAMS positions.

From the figure, it can be seen that the estimated positions for the sUAS did not match the raw ADS-B and LAMS positions as closely as they did for the RV-12. The ADS-B estimated positions initially followed the raw positions, but began to diverge from the raw positions when the GPS signal was artificially degraded. This result was most likely attributed to the Kalman filter weighting associated with the *DynamicKalmanFilterEstimator* coupled with the sharp direction and velocity changes associated with the modified sUAS flight path. These sharp changes likely did not allow enough time for the filter matrices to update, and as a result the estimated positions did not closely match the raw positions. The LAMS results show similar discrepancies, and it can be seen that the estimated LAMS positions do not closely match the raw positions. The presence of position estimate errors with the LAMS data offered further information to support the conclusion that the estimation algorithm could not update quickly enough to follow the sharp direction and velocity changes associated with the updated sUAS flight path. Overall, the estimated positions associated with the LAMS data were more accurate than the estimated positions associated with the ADS-B data, but neither of the estimates accurately reflected the true aircraft positions.

The fused position estimates for the sUAS are shown in Figure 7.10. The figure shows that the fused position estimates more-closely followed the LAMS estimated positions than the ADS-B estimated positions. This result was expected due to the nature of the GPS unit accuracy associated with the sUAS and the presence of GPS degradation and denial during sUAS flight testing. Overall, the fused estimates captured the east-west travel of the sUAS with reasonable accuracy, especially considering the nature of the modified flight path and the associated rapid changes in direction and velocity of the aircraft.

Although the estimation and fusion results associated with the AFSL sUAS did not match the true aircraft positions as closely as the results associated with the RV-12, the fused estimates accurately captured the positions of the sUAS during the flight demonstration. In the future, additional flight tests would be valuable in which the aircraft could be flown around a flight path similar to the planned path detailed in [21]. Overall, during the flight demonstration, the estimation and fusion algorithms proved that reasonably-accurate aircraft position estimates could be generated from the provided ADS-B and LAMS data.

7.3 Flight Test Comparison Data

For the manned aircraft results, the largest error between the true aircraft track and the fused position estimates occurred at the southwest corner of the flight path, where the

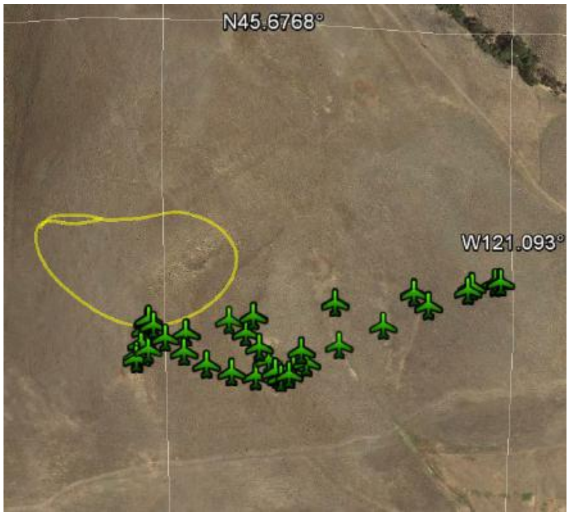


Figure 7.10: sUAS fused (green aircraft) position estimates overlaid on data flash log track (yellow line).

contribution from the wind to flight path deviations was greatest. The difference between the true positions and fused positions at the southwest corner of the flight path was 0.05 NM, which represented a 1% error when compared to the total flight path length of 5 NM. This result was expected due to the fact that GPS information gathered from the RV-12 was not being artificially degraded or denied in any manner.

The sUAS results showed more error than the manned aircraft results, with the greatest difference between the fused estimates and the true aircraft positions being 586 feet, which represented a 158% error when compared to the total flight path length of approximately 370 feet. Several environmental challenges and ADS-B payload challenges could have contributed to this large error, and these issues are discussed in the following section.

7.4 Challenges Encountered

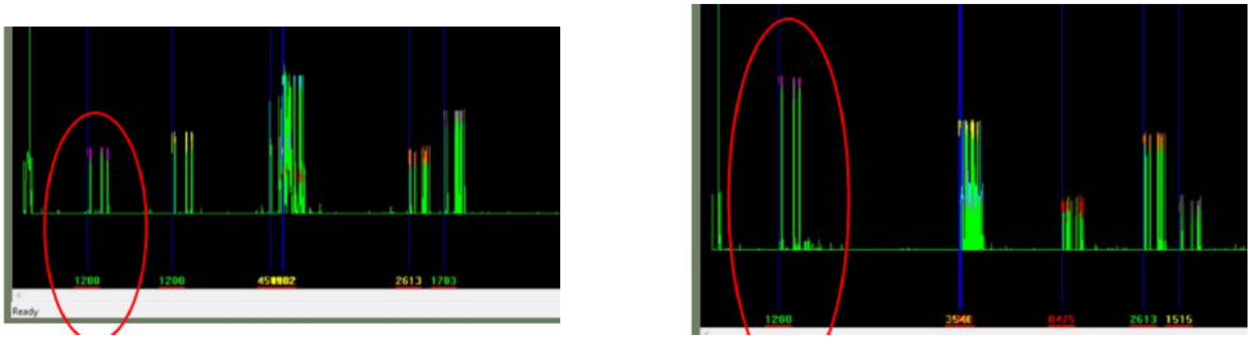
7.4.1 Environmental Factors

During the flight demonstration, environmental factors prevented the original test plan from being accomplished as-desired. On September 22nd, wind speeds at KDLS reached sustained speeds of 25 knots with gusts to 35 knots. Although these conditions did not prevent the operation of the RV-12 and C-172 manned aircraft associated with the test, they did not allow for flight of the sUAS around the desired flight path. On September 23rd when the final test flight of the sUAS was completed, strong winds remained, with sustained speeds of 15 knots with occasional gusts to 20 knots out of the West. Furthermore, intermittent rain was present at the modified test site during the second day of testing. Although weather conditions prevented the sUAS from being flown around the planned rectangular flight path, conditions on the second day allowed for the manual sUAS flight.

7.4.2 Hardware Problems

During the final flight demonstration, it was found that the LAMS was not able to receive a signal from the transponder mounted on the sUAS, but the Clarity ADS-B receiver was able to receive ADS-B information. After further investigation by Karl Winner of ANPC through the use of a hand-held transponder testing kit, it was determined that the transponder being used on the sUAS was not sending the proper pulse widths required by ICAO and FAA requirements. Per these requirements, pulse widths associated with aviation transponders are to be 450 ± 100 nanoseconds. The transponder unit being used on the sUAS was sending out information with measured pulse widths of 345 nanoseconds, and therefore the LAMS was not identifying the signal as a valid transponder. After the first unit was exchanged for a second unit, the LAMS was able to identify and track the transponder, and the measured pulse widths were found to be 375 nanoseconds. A comparison of the signal return strength between the first and second transponder units is shown in Figure 7.11.

From the figure, it can be seen that the strength of the signal corresponding to the sUAS transponder code of 1200 was much greater for the second transponder unit as compared to the first unit. This result was directly related to the compliant pulse width of the second transponder unit. Once the second unit was used, the LAMS was able to track the sUAS and testing continued.



(a) Transponder unit 1 returns.

(b) Transponder unit 2 returns.

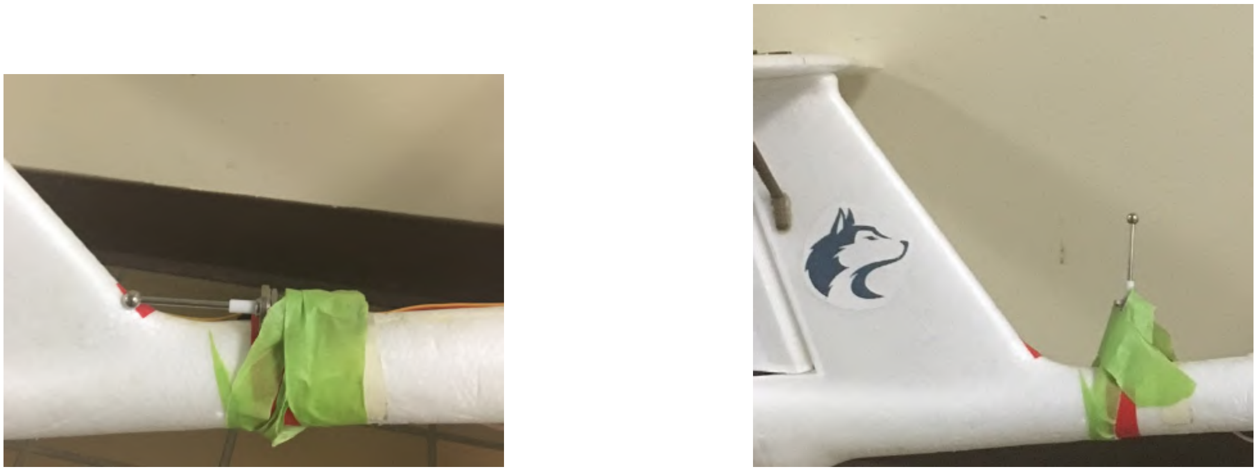
Figure 7.11: Comparison of return strength between first and second transponder units.

7.4.3 EMI and Orientation

Initial testing of the XPS-TR transponder unit in the presence of AFSL sUAS avionics indicated that electromagnetic interference could be an issue. Signals from the transmitter used to control the sUAS are seen at the in-aircraft receiver with a power of 0.1 W, while pulses sent from the transponder are sent at a power of 5 W. This large disparity in power ratings between the transponder and signals used to control the sUAS flight surfaces and associated equipment caused concern for potential loss of aircraft control and potential uncommanded changes in payload operation.

Ground testing of the transponder payload at Meadowbrook Farms and during the KDLS ground test indicated that the transponder often would not respond to changes in mode selection from altitude reporting mode back to standby mode, and this behavior was attributed to electromagnetic interference from the transponder. Additional ground testing showed that despite this supposed interference, all flight control surfaces and the motor on the sUAS remained functional without delays in movement during transponder operation. Furthermore, GPS-related mode changes were accepted by the system regardless of transponder operating state, and seamless change between normal GPS operation, GPS-denied operation, and GPS-degraded operation was observed during all ground testing. In order to further reduce the potential for electromagnetic interference affecting the operation of the payload, aluminum foil was used to wrap all payload connecting wires and line the inside of the Arduino board enclosure.

During initial flight testing of the TRAPIS payload at Meadowbrook Farms, the payload exhibited the same issues, most notably the inconsistent ability of the transponder to switch from altitude reporting mode back to standby mode. Since this issue did not affect the safety of flight and the control of the sUAS, it was not treated as a critical action item or researched further. Initial flight testing showed that all ADS-B Out capabilities of the transponder were working as expected, and during this testing the transponder antenna was oriented on the aircraft fuselage in line with the longitudinal axis as shown in Figure 7.12(a). Due to the orientation of the antenna and the limited payload space afforded with the TRAPIS equipment installed, the antenna ground plane was removed for initial flight testing. The removal of the antenna ground plane did not affect the reception or integrity of ADS-B information, so the ground plane was not re-installed on the aircraft.



(a) ADS-B antenna in original orientation.

(b) ADS-B antenna in modified orientation.

Figure 7.12: Mounting ADS-B near empennage.

In addition to the lack of an adequate ground plane, the orientation of the antenna proved

Figure 7.13: Flight team outside the MFOC with the Skywalker 1900 used for flight testing.

to be an issue. In normal aircraft applications, transponder antennas are oriented vertically either under or on top of the fuselage of the aircraft. This orientation is required due to the vertical polarity of the transponder antennas, and the large size of the aircraft fuselage relative to the antenna wavelength provides a proper ground plane for the antenna. The original transponder antenna position and orientation on the sUAS prevented the antenna from being vertically oriented with respect to the LAMS system. Furthermore, when the nose of the aircraft pointed directly towards or away from the LAMS system, the antenna was oriented in such a way that the signal was lost. This resulted in the LAMS track being continually dropped and regained as the sUAS made turns in the test airspace. In order to remedy this issue, the transponder antenna was placed in a vertical orientation, but still did not have a dedicated grounding plane as shown in Figure 7.12(b). Once the antenna was vertically oriented, the LAMS equipment was able to continuously track the aircraft, and signal was not lost between the aircraft and the LAMS station.

Chapter 8

CONCLUSIONS AND FURTHER RESEARCH

8.1 Flight Testing Conclusions

Based on the results of the ground tests and flight tests, several main conclusions can be drawn. The conclusions should pave the way forward for additional research to be completed as desired by the AFSL and associated industry partners.

Through initial ground testing in the spring of 2016 and flight testing at Meadowbrook Farms and KDLS, it was shown that an ADS-B In transponder could be flown on a commercial off-the-shelf sUAS with consumer-grade avionics while functioning properly. The full transponder payload weighed several pounds, and although the volume pushed the payload capacity limits of a Skywalker 1900 sUAS, the transponder payload package would fit easily in larger sUAS models. Although transponder antenna mounting proved difficult with an adequate ground plane on the aircraft fuselage, ADS-B information was received from the aircraft and the antenna setup was able to be tracked using the LAMS equipment at the KDLS airfield. Additionally, GPS position information provided by the sUAS GPS unit in the ADS-B packets was accurate enough for all required operations, although GPS accuracy could have been increased by using more advanced GPS equipment.

Based on the information gathered from flight test results, it was shown that the LAMS system could serve as a suitable tracking method for sUAS aircraft operating in GPS-degraded or GPS-denied environments. Position information gathered from the LAMS station at KDLS indicated that the LAMS was able to track the sUAS accurately. Comparison of the position information gathered from the ADS-B and LAMS data streams showed that the LAMS position estimates remained close to the ADS-B position estimates within sufficient error bounds. Furthermore, the ability of the LAMS to track the sUAS aircraft operating on a small flight path with high fidelity in difficult conditions proved the utility of the system for primary and secondary tracking purposes. During simulated GPS-degraded and GPS-denied operations, the LAMS continued to track the sUAS appropriately, and the integrity of the LAMS position estimates was not altered by erroneous GPS information. The LAMS demonstrated the ability to track multiple vicinity aircraft at the same time regardless of the GPS integrity associated with each of the aircraft being tracked.

Comparison of the altitudes gathered from the ADS-B and LAMS data streams for the sUAS showed variation that was initially troubling. After analysis, it seems that the differences between the reported GPS and corrected pressure altitudes could be attributed to uncalibrated placement of the pressure altimeter without proper access to an outboard static source and the use of a consumer-level sUAS GPS unit. Additional analysis of the differences between the reported altitudes could provide further insight into possible reasons for the differences. Nonetheless, the differences between GPS altitudes and corrected pressure altitudes seen during flight testing fall within reasonable error bounds.

Although the transponder payload proved reliable and capable of being carried on a commercially-available sUAS, the transponder mode switching functionality should be investigated further. During testing, switching between the standby mode and altitude reporting mode proved inconsistent and difficult. While this difficulty could be attributed to possible electromagnetic interference, mode switches between GPS-denied and GPS-degraded operation of the transponder happened quickly and reliably without delay. In addition to an investigation of the mode switching capability, additional research should be performed to reduce the volume necessary to carry all transponder-related payload components for flight on smaller sUAS.

Based on the position and altitude information provided by the ADS-B transponder through the ADS-B and LAMS data streams, further research should be conducted to investigate ground plane requirements for the transponder antenna. Since aircraft space came at a premium for the Skywalker 1900, the antenna was placed on the aircraft without in-depth consideration of orientation or ground plane requirements. Depending on the sUAS model flown for further testing, preliminary research should be conducted to determine the proper placement of the transponder antenna and associated ground plane. Furthermore, research should be conducted to determine the proper placement of external static pressure ports on any sUAS associated with further testing. Since the pressure altimeter was not plumbed to any aircraft external static pressure port during flight testing, discrepancies between the GPS altitude and corrected pressure altitude arose. In order to correct these discrepancies and narrow down the source of potential altimetry issues, external static pressure port locations should be researched for sUAS airframes involved in further testing.

8.2 Estimation and Fusion Conclusions

In addition to the general conclusions drawn from flight testing, several conclusions can be drawn about the performance of the position estimation and fusion algorithms. Although the algorithms performed as-desired in simulation and in initial flight testing, the flight demonstration proved that additional work should be done to further develop and validate the performance of these algorithms.

Simulations with all of the estimation and fusion algorithms showed that each of the algorithms was best-suited to certain conditions on the ADS-B and LAMS data provided to the TRAPIS software. In applications where no GPS signal degradation or denial is expected, and under the assumption that the LAMS will perform without significant errors, the *DoNothingEstimator* and *SimpleFuser* should be used. As-expected, simulation showed the best results for aircraft tracking when these algorithms were used in the presence of non-degraded position information from both sources. In cases where GPS degradation or denial is experienced, the *DynamicKalmanFilterEstimator* and *KalmanFuser* should be used. Simulation results showed that in the presence of degraded or denied GPS information, these algorithms provided the most accurate estimates of aircraft position to the TRAPIS interface. Under no circumstances should the *KalmanFilterEstimator* be used for tracking of actual aircraft, as it was designed as an incremental step between the *DoNothingEstimator* and the *DynamicKalmanFilterEstimator* and assumes a static GPS error. Similarly, although the

WeightedFuser provided reasonable fused estimates of aircraft position, the *KalmanFuser* provides increased functionality over the *WeightedFuser* and therefore should be used as the primary data fusion algorithm.

Initial flight testing demonstrated the utility of the *DynamicKalmanFilterEstimator* when used with an actual ADS-B data stream. Although a collocated LAMS signal was not available during initial testing, the performance of the estimation algorithm verified the accuracy of position estimates generated for an sUAS flying around a rectangular flight path similar to the one tested in simulation and planned for the final flight demonstration. Future testing should ensure that GPS-degradation is appropriately scaled to the size of sUAS flight paths to ensure that reasonable estimates can be generated when GPS signal is artificially degraded. Furthermore, code associated with artificial GPS degradation should ensure that aircraft velocities correspond to time intervals associated with degraded positions to more-accurately model actual GPS function.

Based on the results of the final flight demonstration, additional testing should be accomplished wherein the sUAS is able to fly around a large, pre-defined flight path similar to the one that was originally planned. The data show that the sUAS was able to be tracked simultaneously by both the Clarity ADS-B In receiver and the LAMS, however the estimates generated from the manual sUAS flight do not realistically demonstrate the full capabilities of the estimation and fusion algorithms. While it is encouraging to know that the sUAS was tracked by both position technologies and that the LAMS track remained reasonably accurate in the presence of artificial GPS degradation and denial, further research is required to demonstrate the full functionality of the estimation and fusion algorithms.

Overall, the use of ADS-B transponders on future sUAS will be dependent upon the size, weight and cost of such transponders. If sUAS are to operate alongside manned aircraft within the NAS while prioritizing safety and situational awareness, ADS-B technologies could provide the means necessary to accomplish such goals. Furthermore, in the event of GPS degradation or denial to localized areas, the use of technologies such as LAMS could provide affected operators with reliable aircraft position information required for the safety of continued flight operations. Although current availability of associated technologies remains cost-prohibitive, additional development could yield viable solutions for further integration of manned and unmanned aircraft into the NAS.

BIBLIOGRAPHY

[1] *Airport and Air Traffic Control System*. Princeton University Press, 1982. URL https://www.princeton.edu/~ota/disk3/1982/8202/820205.PDF.

[2] SparkFun RS232 Shifter - SMD, 2017. URL https://www.sparkfun.com/products/449.

[3] I. 3D Robotics. 3DR uBlox GPS with Compass Kit. URL https://3dr.com/support/articles/207681053/3dr_ublox_gps_with_compass_kit/.

[4] Aircraft Owners and Pilots Association. *Airspace for Everyone*, June 2009.

[5] Arduino AG. Arduino MEGA 2560, 2017. URL https://www.arduino.cc/en/Main/arduinoBoardMega2560.

[6] J. Brandon. Gps jammers illegal, dangerous, and very easy to buy. *Fox News Technology*, 2010.

[7] L. Drolet, F. Michaud, and J. Cote. Adaptable sensor fusion using multiple kalman filters. In *Intelligent Robots and Systems*, February 2000.

[8] Federal Aviation Administration. Equip ads-b: Ads-b airspace, 2016. URL https://www.faa.gov/nextgen/equipadsb/airspace/.

[9] Federal Aviation Administration. Faa aerospace forecast: Fiscal years 2016-2036, 2016.

[10] Federal Aviation Administration. Nextgen: Equip 2020, 2016. URL https://www.faa.gov/nextgen/update/general_aviation/adsb/equip2020/.

[11] Federal Aviation Administration. *Pilot's Handbook of Aeronautical Knowledge*, 2016.

[12] Federal Aviation Administration. Fact sheet - small unmanned aircraft regulations (part 107), June 2016. URL `https://www.faa.gov/news/fact_sheets/news_story.cfm?newsId=20516`.

[13] Federal Aviation Administration. Ifr operations in the national airspace system, 2017. URL `http://web.ics.purdue.edu/~dsun/documents/atc_faa.pdf`.

[14] Federal Register. *14 CFR 91.131 - Operations in Class B Airspace*.

[15] GeoMidpoint. Geographic midpoint calculation methods, 2017. URL `http://www.geomidpoint.com/calculation.html`.

[16] W. Handley. Two nextgen air safety tools: An ads-b equipped uav and a wake turbulence estimator. Master's thesis, University of Washington, Seattle, WA, June 2016.

[17] jDrones. Ardupilot. URL `http://firmware.us.ardupilot.org/`.

[18] C. Lai, Y. Ren, and C. Lin. Ads-b based collision avoidance radar for unmanned aerial vehicles. In *Microwave Symposium Digest*, 2009.

[19] M. C. Lee. Ads-b buyer's guide 2016 final countdown, February 2016. URL `http://www.planeandpilotmag.com/article/ads-b-buyers-guide-final-countdown/#.WJ-TyW8rJbU`.

[20] M. Lorenz. Px4 autopilot. URL `https://pixhawk.org/`.

[21] C. Lum, R. Larson, W. Handley, S. Lui, and Z. Caratao. Flight testing an ads-b equipped suas in gps-denied environments. 2017.

[22] C. W. Lum and B. Waggoner. A risk based pradigm and model for unmanned aerial vehicles in the national airspace. In *Proceedings of the 2011 Infotech@Aerospace Conference*, St. Louis, MO, March 2011.

[23] C. W. Lum, K. Gauksheim, T. Kosel, and T. McGeer. Assessing and estimating risk of operating unmanned aerial systems in populated areas. In *Proceedings of the 2011 AIAA Aviation Technology, Integration, and Operations Conference*, September 2011.

[24] F. Martel, R. Shultz, S. W, Z. Wang, and M. Czarnomski. Unmanned aircraft systems sense and avoid avionics utilizing ads-b transceiver. In *AIAA Aerospace Conference*, 2009.

[25] S. Mohleji and G. Wang. Modeling ads-b position and velocity errors for airborne merging and spacing in interval management application. Technical report.

[26] R. Niles. Gps interference notam for southwest. *AvWeb*, 2016.

[27] S. Ramasamy, R. Sabatini, and A. Gardi. Avionics sensor fusion for small unmanned aircraft sense-and-avoid. In *Meteorology for Aerospace*, 2014.

[28] A. Ross. Pilots report four drone near misses in a month in uk, November 2016. URL `https://www.theguardian.com/technology/2016/nov/17/drone-came-within-5-metres-of-passenger-jet-at-liverpool-airport`.

[29] Sagetech Corporation. XP Family of Transponders, 2017. URL `https://sagetech.com/xp-transponders/`.

[30] A. Schofield. Nextgen breakeven shifts to 2020, faa says. *Aviation Week*, 2012.

[31] M. Scislowska. Drone seized in Poland, man questioned after near-collision with passenger plane, 2015. URL `https://www.suasnews.com/2015/07/drone-seized-in-poland-man-questioned-after-near-collision-with-passenger-plane/`.

[32] B. Stark, B. Stevenson, and Y. Chen. Ads-b for small unmanned aerial systems: Case study and regulatory practices. In *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2013.

[33] K. K. Ueunten, C. W. Lum, A. A. Creigh, and K. Tsujita. Conservative algorithms for automated collision awareness for multiple unmanned aerial systems. In *Proceedings of the 2015 IEEE Aerospace Conference*, March 2015.

[34] P. Zarchan, H. Musoff, and F. Lu. *Fundamentals of Kalman Filtering: A Practical Approach.* AIAA, February 2009.