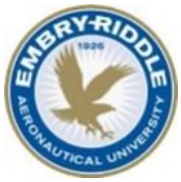


A11L.UAS.86 - A44 Mitigating GPS and ADS-B Risks for UAS

Task 2: Identification of Potential Mitigations

August 5, 2022

NOTICE

This document is disseminated under the sponsorship of the U.S. Department of Transportation in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof. The U.S. Government does not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the objective of this report. The findings and conclusions in this report are those of the author(s) and do not necessarily represent the views of the funding agency. This document does not constitute FAA policy. Consult the FAA sponsoring organization listed on the Technical Documentation page as to its use.

LEGAL DISCLAIMER

The information provided herein may include content supplied by third parties. Although the data and information contained herein has been produced or processed from sources believed to be reliable, the Federal Aviation Administration makes no warranty, expressed or implied, regarding the accuracy, adequacy, completeness, legality, reliability or usefulness of any information, conclusions or recommendations provided herein. Distribution of the information contained herein does not constitute an endorsement or warranty of the data or information provided herein by the Federal Aviation Administration or the U.S. Department of Transportation. Neither the Federal Aviation Administration nor the U.S. Department of Transportation shall be held liable for any improper or incorrect use of the information contained herein and assumes no responsibility for anyone's use of the information. The Federal Aviation Administration and U.S. Department of Transportation shall not be liable for any claim for any loss, harm, or other damages arising from access to or use of data or information, including without limitation any direct, indirect, incidental, exemplary, special or consequential damages, even if advised of the possibility of such damages. The Federal Aviation Administration shall not be liable to anyone for any decision made or action taken, or not taken, in reliance on the information contained herein.

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. Enter the report number assigned by the sponsoring agency.	**2. Government Accession No.**	**3. Recipient's Catalog No.**
4. Title and Subtitle A11L.UAS.86 - A44 Mitigating GPS and ADS-B Risks for UAS Task 2: Identification of Potential Mitigations		**5. Report Date** August 5, 2022 **6. Performing Organization Code**
7. Author(s) University of North Dakota William Semke, william.semke@und.edu Prakash Ranganathan, prakash.ranganathan@und.edu Kansas State University Randall Nichols, profrknichols@ksu.edu Embry-Riddle Aeronautical University Hever Moncayo, moncayoh@erau.edu Oregon State University Jihye Park, jihye.park@oregonstate.edu		**8. Performing Organization Report No.** Enter any/all unique alphanumeric report numbers assigned by the performing organization, if applicable.
9. Performing Organization Name and Address University of North Dakota 243 Centennial Dr. Grand Forks, ND 58202		**10. Work Unit No.** **11. Contract or Grant No.**
12. Sponsoring Agency Name and Address FAA		**13. Type of Report and Period Covered** Task 1 **14. Sponsoring Agency Code** code

15. Supplementary Notes

Enter information not included elsewhere, such as translation of (or by), report supersedes, old edition number, alternate title (e.g. project name), hypertext links to documents or related information in the form of URLs, PURLs (preferred over URLs - https://purl.org/docs/index.html), DOIs (http://www.doi.org), insertion of QR codes, copyright or disclaimer statements, etc. Edit boilerplate FHWA statement above if needed.

16. Abstract

This report on the Identification of Potential Mitigations report fulfills Task 2 for the A44 ASSURE project. Recorded ABS-B data was analyzed to reveal dropouts and anomalies that occur in flight operations. The performer conducted a market survey of market solutions to mitigate loss of GPS and loss of ADS-B data as well as a market survey of market solutions to mitigate unvalidated GPS and unvalidated ADS-B In data. The market surveys include estimated costs, ease of implementation, and a preliminary assessment of the effectiveness of market solutions to mitigate the various risks identified in Task 1.

17. Key Words GPS, ADS-B, signal dropouts, erroneous data, jamming, spoofing	**18. Distribution Statement** No restrictions. This document is available through the National Technical Information Service, Springfield, VA 22161. Enter any other agency mandated distribution statements. Remove NTIS statement if it does not apply.

19. Security Classification (of this report) Unclassified	**20. Security Classification (of this page)** Unclassified	**21. No. of Pages** 61	**22. Price**

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

TABLE OF ACRONYMS

1090ES	1090 Extended Squitter Data Link
a/c	Aircraft (Piloted or unmanned) also A/C
A/CFD	Aircraft Flood Denial jamming
ACAS	Airborne Collision Avoidance System
ADS - B	Automatic Dependent Surveillance – Broadcast systems
AOA	Angle of Arrival of signals to GPS receivers
ATC	Air Traffic Control / Air traffic Control Signals
ATCC	Air Traffic Control Center
ATM	Air Traffic Management
ATRAN	Automatic Terrain Recognition and Navigation System
ATS	Air Traffic Services
ATSAW	Air Traffic Situational Awareness
ATO	Air Traffic Organization
BVLOS	Beyond Visual Line-Of-Sight operations
C2	Command and Control
C /A	GPS Satellite Course Acquisition unique code, $ca_I(t)$ in Appendix A
C/No	Carrier to Noise ratio
CCC	Circular Cross-Correlation
CD	Collective Detection maximum likelihood localization approach (Eichelberger 2019)
CDMA	Code Division Multiple Access Protocol
CIA	Confidentiality, Integrity & Availability (standard INFOSEC paradigm)
CM	Countermeasure
CNPC	Control and Non-Payload links
COTS	Commercial Off-The-Shelf
CTN	Course -Time Navigation
DHS	Department of Homeland Security
DoS	Denial-of-Service
ECD	Dr. Manuel Eichelberger's advanced implementation of CD to detect and mitigate spoofing attacks on GPS or ADS-B signals (Eichelberger 2019)
ERAU	Embry Riddle Aeronautical University
FAA	Federal Aviation Administration
GCS	Ground Control Station
GNSS	Global Navigation Satellite System (GPS,GLONASS, Galileo, Beidou & other regional systems)
GNU	GNU / Linux Operating system
GPS	Global Positioning System
GS	Ground Station
GSFD	Ground Station Flood Denial jamming
HAPS UAVs	UAVs dedicated to HAPS service (example to communicate via CNPC links)
HAPS	High Altitude Platforms (generally for wireless communications enhancements)
HOW	Hand-Over-Word satellite data timestamp defined in (I.S.-G.P.S.-200G 2013)
ICAO	International Civil Aviation Organization
IMU	Inertial Measurement Unit
INFOSEC	Information Security
INS	Inertial Navigation System
ITE	Installation, Training & Expensive

ITP	In Trail Procedure
KSU	Kansas State University
LED	Light Emitting Diodes
LOS	Line-of-sight / Loss of Signal / Loss of Separation
LTE	Long-Term Evolution
MitM	Man-in-the-Middle
MLAT	Multilateration System
NASA	National Aeronautics and Space Administration
NDM	Navigation data modification spoofing attack
NLSO	Non-Line-Of-Sight
NMA	See Navigation Message Authentication
OrSU	Oregon State University
OTH	Over-The-Horizon
PEN	Probabilistic Environment Navigation
PMU	Phasor Measurement Unit
PRN	Pseudo-Random Noise
PSR	Primary Surveillance Radar
RF	Radio Frequency
RFID	Radio Frequency Identification (tags)
RN	Ryan-Nichols qualitative information security risk equations
RSS	Received Signal Strength.
SDR	Software Defined Radio.
SEN	Structured Environment Navigation
SIC	Successive Signal Interference Cancellation
SLAM	Simultaneous Localization and Mapping
SNR	Signal to Noise Ratio
SRMGSA	Safety Risk Management Guidance for System Acquisitions
SSLT	Seamless Satellite-Lock Takeover spoofing attack
SSR	Secondary Surveillance Radar
sUAS	Small Unmanned Aircraft System
TCAS	Traffic Collision Avoidance System
TDOA	Time Difference Of Arrival
TOA	Time of Arrival
ToF	Time of Flight
TTFF	Time To First Fix (latency)
UAF	University of Alaska, Fairbanks
UAS	Unmanned Aircraft Systems
UAV	Unmanned Aerial Vehicle
UAT	Universal access transceiver
UND	University of North Dakota
USAF	United States Air force
UTM	Unmanned Traffic Management
VDL	VHF Data link
WAM	Wide Area Multilateration

EXECUTIVE SUMMARY

Unvalidated or unavailable Automatic Dependent Surveillance-Broadcast (ADS-B) and Global Position Systems (GPS) data poses security and safety risks to automated Unmanned Aircraft Systems (UAS) navigation and to Detect and Avoid (DAA) operations. Erroneous, spoofed, jammed, or drop outs of GPS data may result in unmanned aircraft position and navigation being incorrect. This may result in a fly away beyond radio control, flight into infrastructure, or flight into controlled airspace. Erroneous, spoofed, jammed, or drop outs of "ADSB-In" data may result in automated unmanned aircraft being unable to detect and avoid other aircraft or result in detecting and avoiding illusionary aircraft. For automated DAA, a false ADS-B track can potentially be used to corral the unmanned aircraft to fly towards controlled airspace, structures, terrain, and so on. This research is necessary to enable safe and secure automated small UAS (sUAS) navigation and safe and secure automated sUAS DAA operations. Goals for the project include reports and recommendations useful for Federal Aviation Administration (FAA) policy development and UAS standards development. It is expected that this information will be used to better understand the risks and potential mitigations, and to help the FAA to reassess and refine FAA policy with respect to validation of ADS-B data.

The A44 team has completed the Identification of Potential Mitigations report which fulfills Task 2 for the A44 ASSURE project. Recorded ABS-B data was analyzed to expose potential risks and provide guidance on mitigation schemes. The examination reveals drop outs and anomalies that occur in flight operations. Based on the risk assessments in Task 1, the performer conducted a market survey of market solutions to mitigate loss of GPS and loss of ADS-B data as well as a market survey of market solutions to mitigate unvalidated GPS and unvalidated ADS-B In data. The market surveys include estimated costs, ease of implementation, and a preliminary assessment of the effectiveness of market solutions to mitigate the various risks identified in Task 1.

The integrity of ADS-B and GPS navigation systems was analyzed to detect threats to the integrity and/or reliability of the data. These risks include erroneous, spoofed, jammed, and dropped data from GPS and/or ADS-B systems. Recorded ABS-B data was examined to expose potential risks and provide guidance on mitigation schemes. Two primary pre-recorded data set types were used in this study; GPS data from the Dallas Fort Worth Airport and data from the OpenSky Network. The results are informative and provide real-world assessment of GPS and ADS-B navigation data.

Several mitigation schemes were evaluated for their effectiveness in jamming and spoofing conditions. The mitigation schemes evaluated were optical flow, geomagnetic navigation, cellular signal navigation, Wi-Fi navigation, and the Eichelberger's Collective Detection (ECD) method. The results from these five systems indicate that most have an overall high effectiveness rating, while having varying effectiveness in the individual factors scored. It should be noted that additional mitigation strategies were briefly reviewed but were not of sufficient interest by the team to include in the full evaluation.

It is the A44 team opinion that flight and simulation testing should continue on all five of the mitigation methods and continued efforts be made in identifying drop outs and erroneous data in the current data sets along with new data sets obtained. These efforts will be summarized in the A44 Task 3 Planning the Testing and Demonstration of Mitigations report.

I. INTRODUCTION & BACKGROUND

The FAA position communicated to RTCA Special Committee 228 is that UAS DAA systems should validate "ADS-B In" data before it is used to conduct Detect and Avoid (DAA). A risk assessment and exploration of potential solutions is needed to inform potential policy updates for different types of UAS and operations for both GPS validation and ADS-B In validation. Potential risks and/or mitigations examples considered at the onset of the project are listed below.

- Potential Risk: If GPS data drops out or is jammed, the UAS may not know exactly where it is located and may fly away without anyone's knowledge of where it is. Note that sUAS are not tracked by Air Traffic Control (ATC) radar. Potential mitigations include means to detect broad area GPS jamming or GPS dropouts. Examples: monitor the known GPS position of a fixed GPS receiver on a cell phone, ground control station, tower, and other UAS that is on the ground. Alternatively, have an independent means of temporary navigation and UAS tracking sufficient to cease operations safely. Examples: Inertial Measurement Unit (IMU) navigation, UAS beacons (Radio Frequency (RF) or optical), vision-based navigation, rough triangulation or signal direction finding from the ground using Command and Control (C2) Signal to Noise ratio or time of flight analysis, etc.
- Potential Risk: If GPS signals are spoofed, the UAS may think it is in one location when it is actually in another location. This may result in the UAS crossing airspace boundaries, flying beyond radio control, sudden climbing to avoid terrain referenced onboard digital terrain elevation maps, etc. Potential mitigations could include means to detect broad are GPS spoofing. Examples: monitor the known GPS position of a fixed GPS receiver on a cell phone, Ground Control Station (GCS), tower, or other UAS that is on the ground. Alternatively, have an independent means of temporary navigation sufficient to cease operations. Potential examples may include: temporary IMU navigation, navigate by C2 signal strength, UAS beacons (RF or optical), vision-based navigation, etc.
- Potential Risk: "ADS-B In" signals drop out or are jammed. This prevents UAS from detecting and avoiding other aircraft that are transmitting "ADS-B Out". Potential mitigations could include a means to detect ADS-B dropouts and jamming to cease UAS operations when jamming is detected. Example: monitor the signal from a fixed "ADS-B Out" source (potentially easy and low cost). Alternatively, potential mitigations could rely upon detecting jamming, have a means to safely cease DAA operations.
- Potential Risk: A false "ADS-B In" signal is detected that harasses the UAS. If the UAS is automated to avoid collisions with other aircraft, there is the potential for false signals to harass and corral an automated UAS thereby directing it where a malicious actor desire it to fly (fly into infrastructure, terrain, controlled airspace, etc.). Potential mitigations could include having a means to validate "ADS-B In" tracks or detect false tracks. Example solutions: rough triangulation or signal direction finding from the ground using Signal to Noise ratio or time of flight analysis. Have an ability for overriding UAS automated collision avoidance on unvalidated "ADS-B In" tracks. Cease UAS operations when false (ADS-B In" tracks are detected.

This project assesses the safety and security risks of unvalidated GPS and ADS-B In data used to support a variety of UAS operations to include primarily sUAS operations, while also providing data to unmanned cargo transport and remotely piloted passenger transport

operations where applicable. For sUAS operations, low cost and easy to implement mitigations commensurate with their safety and security risks, and are therefore is emphasized.

Based on the risk assessment in Task 1, the performer conducted a market survey of available solutions to (1) mitigate loss of GPS and loss of ADS-B data as well as (2) unvalidated GPS and unvalidated ADS-B In data. The market surveys include estimated costs, ease of implementation, and a preliminary assessment of the effectiveness of market solutions to mitigate the various risks identified in Task 1 for the various UAS operations. The performer also investigated other potential methods, operational mitigations, strategic mitigations, or other means for addressing potential safety and security risks that were not identified through the market survey. These additional mitigations were assesses with the same criteria as the market survey to mitigate the risks identified in Task 1.

GPS mitigation strategies for denied and/or jammed environments were explored and potential solution proposed. Cybersecurity and counterintelligence measures were investigated to decrease the risk of disruption or takeover. Examination of recorded ABS-B data was conducted to expose potential risks and provide guidance on mitigation schemes. The examination reveals drop outs and anomalies that occur in flight operations. While the exact cause may not be initially known, insight will be obtained to better focus on the most likely causes. These strategies will also be assessed for their cost, ease of implementation, and ability to mitigate the risks identified in Task 1.

II. Risk Assessment of Potential Mitigations

The mitigation strategies identified were evaluated using an assessment tool to provide a metric to the overall effectiveness. The proposed assessment metrics assess the overall effectiveness of mitigation schemes. Five things are evaluated to quantify the overall score to rank the proposed methods. These factors are:

1.) Cost

2.) Technical Readiness

3.) Ease of Implementation/Use

4.) Size, Weight, and Power (SWaP)

5.) Impact

Each factor will be ranked with a numerical score from 1 to 5, with 1 being the "worst" and 5 being the "best" in each category. A detailed guide for each ranked factor is provided based on the effectiveness of the implementation of the mitigation scheme on a small UAS. Therefore, the factors are the added impact on the "standard' operating configuration.

Cost Rankings

1- Cost over $1000

2- Cost between $500 to $1000

3- Cost between $250 to $500

4- Cost between $100 to $250

5- Cost under $100

Technical Readiness

1- Concept phase

2- Initial prototype testing underway

3- Prototype testing completed

4- Experimental version available

5- Commercially available

Ease of Implementation/Use

1- Extensive modifications/training required

2- Major modifications/training required

3- Moderate modifications/training required

4- Minor modifications/training required

5- Minimal modifications/training required

Size, Weight, and Power (SWaP)

1- Weight greater than 1 kg or power greater than 1 kW

2- Weight between 100 g to 1 kg and/or power between 100 W to 1 kW

3- Weight between 10 g to 100 g and/or power between 10 W to 100 W

4- Weight between 1 g to 10 g and/or power between 1 W to 10 W

5- Weight less than 1 g and power less than 1 W

Impact

1- No impact

2- Little impact

3- Moderate impact

4- Major impact

5- Extensive impact

The cumulative score of the ranked factors will generate a value that is indicative to the overall effectiveness. Each factor in the total score has an equal weighting and the sum of all ranking produce the overall score. A scoring breakdown is color coded to outstanding, high, medium, or low value to indicate the overall effectiveness as shown in Table 1.

Table 1. Potential mitigation effectiveness scoring system

Score	Effectiveness
5-10	Low
10-15	Medium
15-20	High
20-25	Outstanding

The scoring system provides a numerical score to aid in overall effectiveness, however this score is to be used for a guide to aid in identifying mitigation strategies with high effectiveness in the current state of development. Some mitigation strategies may have great potential but are early in their development. These strategies, while perhaps do not score high at this time, may have the potential to have a great impact with further development.

III. UAS Navigational Anomalies – Dropouts and Erroneous Data Potential Mitigations Assessment

Messages broadcast failures, often referred to as "dropouts," are common in unmanned aerial vehicles and can occur when a receiver fails to receive messages over time (Semke et al. n.d.; Tabassum and Semke 2018a; Shaukat et al. n.d.). In general, time instance of 'last contact' of flight is tracked to determine how often such message dropout can occur and this can vary across aircraft or small-scale unmanned aerial system types. Causes of message dropout are unknown and challenging to point out, as several factors could be responsible for the loss of message or GPS/ADS-B packets. Some potential causes include 1) high terrain environments; 2) intentional jamming (e.g., at locations such as air force bases and critical infrastructures) ;3) fading phenomena. For example, fading in wireless systems can lead to multi-path induced distortion, which can affect certain UAS components (GPS, IMUs, or receivers). and it is important for operators on the ground to know when a dropout has occurred, as this information is routinely used for detect-and-avoid (DAA) procedures (Semke et al. n.d.). The integrity of navigation systems, such as ADS-B and GPS, must be analyzed to detect anything that threatens their integrity. UAS relies on a satellite infrastructure that provides positioning, navigation, and timing capabilities. GPS/ADS-B are critical sensors for realizing such capabilities, but these systems offer functions far beyond just navigation (U.S. Department of Transportation 2022a; 2022b). The FAA A44 Research Project aims to develop methods to detect and mitigate GPS and ADS-B risks for unmanned or autonomous aircraft systems. These risks include erroneous, spoofed, jammed, and dropped data from GPS or ADS-B systems. These risks cause aircraft navigation problems and can allow bad actors to control the aircraft, causing it to fly to unintended and potentially hazardous locations. Unmanned aircraft may also have trouble detecting and avoiding other aircraft, resulting in collisions.

An assessment of a mitigation scheme using an artificial intelligence path prediction algorithm to aid navigation during periods of ADS-B/GPS dropout/erroneous detection was done. The following are the rankings and brief description on how they were made.

- Cost – Rank 3

Futuristic algorithms can easily run on portable embedded hardware, and thus may come with pre-loaded mitigation software to address drop out or erroneous data handling. The team envisions aviation industry will design potential deep-learning or machine learning application that could be cost-effective, can process real-time flight data on board computer or may use a cloud-based infrastructure for data transfer, storage, and to process continuous streaming data parameters (latitude, longitude, altitude, or other positional/navigational related parameters.

- Technical Readiness – Rank 1

Currently, there is no software on the market to detect dropout rates or erroneous data handling, this project is still in its early concept phase, requiring end-to-end code development as well as Software Quality Testing prior to packaging and release.

- Ease of Implementation/Use - Rank 3

A miniature hardware board (e.g., Raspberry Pi, Jetson Nano, or similar device) can be mounted on any UAS and bundled with machine learning application to detect the dropouts in UAS.

- Size, Weight, and Power (SWaP) – Rank 2

The hardware component, such as the *Raspberry Pi 4*, weighs approximately 46 grams and consumes approximately 5-15 watts., so rank 2 is appropriate for running any developed algorithms that may weigh under 200 grams.

- Effectiveness – Rank 4

The implementation of detecting dropout and erroneous data will assist path planning and address navigational problems, so we choose rank 4, as this solution may result in major impact.

Assessment Score: 13-Medium

The next three sections examine recorded ADS-B data to expose potential risks and provide guidance on mitigation schemes. It reveals dropouts and anomalies that occur during flight operations and provides critical information regarding the likelihood and extent of drop outs and/or erroneous data. While the exact cause may not be initially known, insight will be obtained to better focus on the most likely reasons. This study used multiple data set types: GPS data from the Dallas Fort Worth Airport (DFW), Alaska, and ADS-B data from the OpenSky Network. The results are detailed in three sections:

- Section 1: Outlier Based GPS Dropout Detection in DFW Airport small-scale unmanned aerial Dataset
- Section 2: Detection of ADS-B / GPS dropout using OpenSky Network Data. This section has two approaches for detection: 1) Z-score/ensemble scoring analysis; 2) NIC, NAC based criteria for validating GPS integrity check
- Section 3: Imputation of GPS Dropout points using Machine Learning Models for OpenSky Network

III.1. OUTLIER BASED GPS DROPOUT DETECTION

III.1.1. ADS-B/GPS Dropout Detection on UAS Flights at Dallas Forth Worth (DFW)

This study develops a statistical framework for identifying an upper bound for acceptable time delays between consecutive GPS messages. Time delays greater than this bound are considered instances of message dropout. It is anticipated that this methodology can easily be extensible to ADS-B data sets in addition to the GPS data presented here.

GPS data were collected over 24 hours for UASs in the range of a telemetry receiver located at the Dallas Fort-Worth (DFW) Airport. Since many UASs only briefly entered the area over which signals could be received, the entire data set was initially filtered for flights containing at least 100 messages. This criterion provided flights with enough data to perform statistical analysis on the time delay between consecutive messages of unique flights, while flights containing fewer messages were discarded.

Since the recorded flights were for random UASs entering the monitored region, it is difficult to establish what the actual time delay between the transmission of consecutive messages is for

a particular flight as there is no control data set with which to compare, and there is no way of accessing the onboard computer of the UASs to retrieve this data. Therefore, this work aims to estimate this time delay statistically from the messages received and determine an appropriate upper bound for this time delay. Consecutive messages separated by a time delay greater than this upper bound are considered instances of message dropout.

Statistical outliers for the time delay between consecutive messages were defined as any time delay greater than the mean time delay ($\widehat{\Delta}t$) plus the root-mean-square-error (Δt_{RMSE}) of the time delay multiplied by a constant $k = 2$, as shown in Eq. 1.

$$\text{Outlier}_{min} = \widehat{\Delta}t + k \times \Delta t_{RMSE} \quad (1)$$

where Δt_{RMSE}, is defined as in Eq. 2,

$$\Delta t_{RMSE} = \sqrt{\sum_1^N \left(\Delta t_i - \widehat{\Delta}t\right)^2} \quad (2)$$

Note that a value of 2 was chosen for k since smaller values removed all but the smallest Δt and larger values prevented large time delays (relative to the mean) from being classified as outliers. The values $\widehat{\Delta}t$ and Δt_{RMSE} were then recalculated with all detected outliers removed from the data set for a given flight. This procedure was repeated iteratively until no outliers remained and the final resulting upper bound for determining an outlier in the set was considered the minimum time delay between consecutive messages to be considered an instance of message dropout.

The above procedure was implemented in Python which can be found at Github Repository where "dfw.py" identifies all flights in the data set containing more than 100 messages, and flightAnalysis.py performs the iterative outlier removal and defines a time span threshold used to identify dropout instances for each of these flights.

The data set under analysis contained 33 flights that contained at least 100 messages. A summary of the data recorded for these flights is shown in Table 2.

In Table 2, it can be seen that there were a variety of drones detected during the data recording interval. Also note how then maximum altitude varies between flights. Fig. 1 shows a box plot for the flight time, the maximum altitude, the filtered mean time interval (time delay between consecutive messages), and the upper bound of the RMSE filter (time delays above this value for a particular flight are considered dropout instances).

Table 2. Summary of the 33 UAS flights containing more than 100 messages

Flight ID	**Drone Type**	**Flight Time (h:m:s)**	**Upper Bound of RMSE Filter (seconds)**	**Filtered Mean Time Interval (seconds)**	**Filtered Mode Time Interval (seconds)**	**Filtered Median Time Interval (seconds)**
36	P4 Series	0:17:49	5.030052	2.974843	2	2
130	Mavic Mini 2	0:14:10	6.322372	3.627907	4	4

171	Mavic Mini 2	0:21:22	6.633531	3.829268	4	4
199	Mavic Mini 2	0:15:26	6.248761	3.496815	4	4
205	Mavic Air 2	0:06:38	6.276918	3.468085	4	4
225	Mavic Mini 2	0:14:33	4.641503	2.709091	2	2
247	Mavic Air 2	0:18:59	6.207391	3.776786	4	4
282	Mavic Mini 2	0:22:28	6.238912	3.565574	4	4
295	Mavic Mini 2	0:13:50	5.027758	3.064516	4	4
487	Mavic Air 2	0:13:39	4.779035	2.813953	2	2
625	Mavic Mini 2	0:20:24	5.0611	2.958333	2	2
644	Mavic Air 2	0:12:56	12.0737	5.482353	2	5
685	Mavic Mini 2	0:12:24	6.141757	3.457447	4	4
829	M300 RTK	0:17:01	5.178542	3.263889	4	4
855	M300 RTK	0:13:18	6.200319	3.650485	4	4
917	Mavic 2	0:15:13	6.675467	3.569444	4	4
994	Mavic Air 2	0:16:26	4.705989	2.724359	2	2
1025	Mavic Mini 2	0:15:43	6.351594	3.493506	4	4
1041	Mavic Mini 2	0:15:51	5.087642	3.084211	4	4
1154	M300 RTK	0:22:29	6.413206	3.807229	4	4
1172	Mavic 2	0:21:35	6.050304	3.591241	4	4
1289	Mavic Air 2	0:12:16	5.18492	3.083333	4	4
1296	Mavic Mini	0:18:25	6.137266	3.257353	2	3
1344	Mavic Mini 2	0:14:00	6.278763	3.333333	2	4
1398	Mavic Mini 2	0:16:21	10.18167	4.820755	2	4
1412	Mavic Mini	0:18:33	6.99233	3.81203	2	4
1526	Mavic Air 2	0:13:48	6.331115	3.51269	4	4
1617	Mavic 2	0:21:29	8.271192	4.654412	4	4
1689	M300 RTK	0:26:41	6.466892	3.605769	4	4
1695	Mavic Mini 2	0:08:18	6.117624	3.373626	4	4
1731	FPV	0:11:23	6.378759	3.62037	4	4
1738	Mavic Mini 2	0:07:04	4.920513	2.935484	2	2
1740	M300 RTK	0:25:51	6.299064	3.573643	4	4

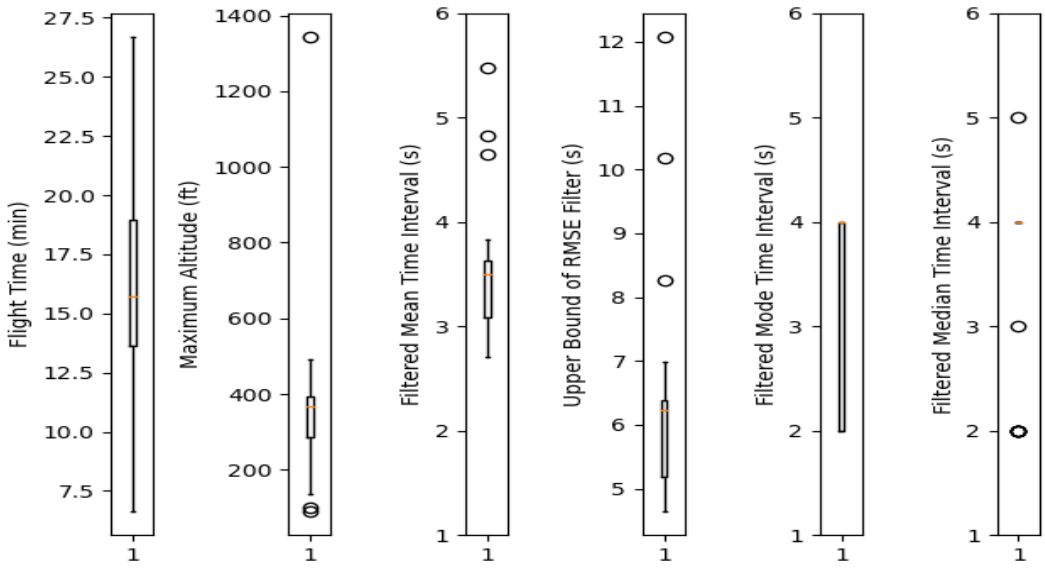


Figure 1. Statistical results for all 33 flights analyzed.

Fig. 1 shows that all flight durations were less than 30 min with an average flight time near 16 min. The average maximum altitude was near 375 ft, and a typical time delay after outlier removal was approximately 3.5 seconds. The average upper bound for determining dropout instances was near 6.25 seconds for the flights analyzed.

A sensitivity analysis is to be performed on the value of k used to define the minimum threshold of an outlier. The detection of outliers has been seen to be highly dependent on this value and it is intended to identify the optimal value of this parameter for defining dropout instances. Additionally, the median and mode of the time delay will be used (in place of the mean) for detecting dropout instances to see how they compare with the current method.

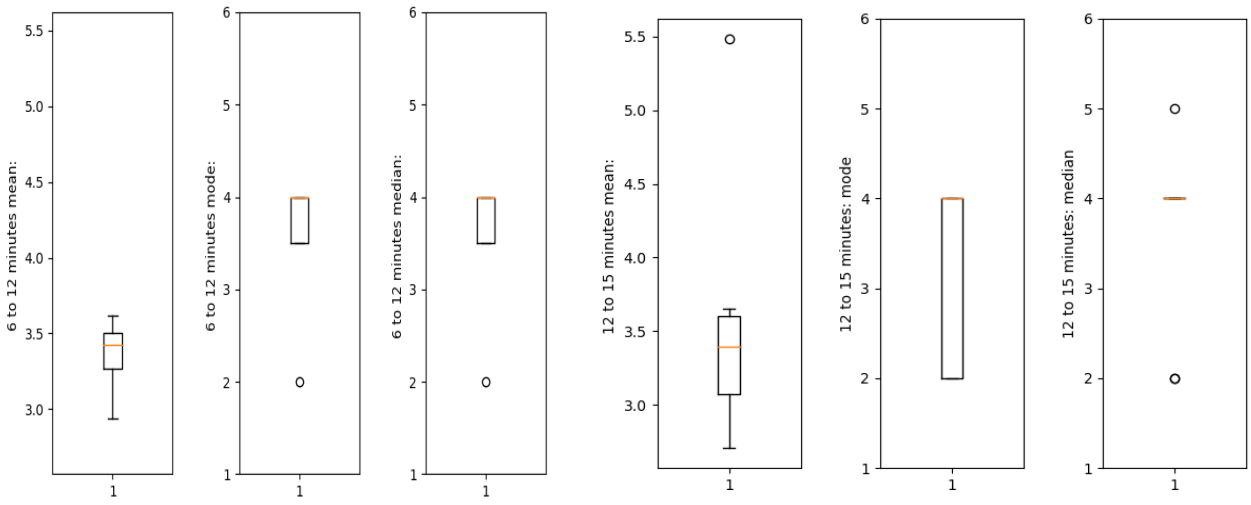
For example, Table 3 illustrates that for Mavic mini 2, the average number of messages reported per 1-min window can vary from 11-18; 14-27 for every two minutes and 20-52 for every three minutes.

Table 3 displays the mean calculations for each of the 33 flights that has at least 100 entries. There is a common trend showing that the aggregate values decline slightly when expanding to 2 minutes and 3 minutes rolling mean analysis. The overall highest aggregate is with the M300 RTK drone with a total of 9.54. The greatest drop between intervals can be seen within the Mavic Mini series, dropping from 8.21 seconds to 6.68 seconds from the 1-minute mean to the 2 minute mean.

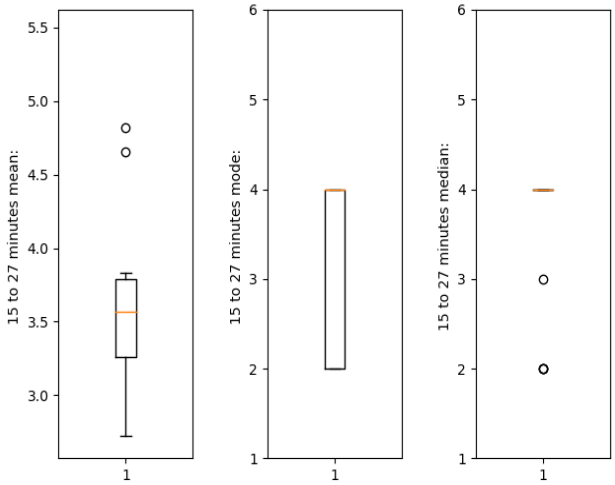
Table 3. Dynamic Rolling Mean window analysis for detecting message reporting frequency.

Drone Type	**Num. of Flights**	**Flight Time**	**1 Minute Mean/Average No. of Messages**		**2 Minute Mean/Average No. of Messages**		**3 Minute Mean/Average No. of Messages**		**Total No. of Messages**
P4 Series	1	0:17:49	3.2686	19.7	3.2645	37.2	3.2667	55.8	335
Aggregate Value			**3.2686**	**19.7**	**3.2645**	**37.2**	**3.2667**	**55.8**	**335**
	2	0:14:10	5.7624	11.4	5.7653	22.7	5.7381	31.8	159

Mavic Mini 2	3	0:21:22	5.5013	12.4	5.4153	22.4	5.2956	30.9	247
	4	0:15:26	4.9929	12.7	4.9121	23.9	4.9250	31.8	191
	5	0:14:33	7.0531	11.3	6.3307	22.6	5.7528	31.6	158
	6	0:22:28	5.2742	13.0	4.9086	25.9	4.7977	35.6	285
	7	0:13:50	9.8383	8.0	9.2437	14.7	9.4028	20.6	103
	8	0:20:24	3.3522	18.4	3.3446	36.7	3.3473	52.4	367
	9	0:12:24	6.5980	10.8	6.9181	21.5	6.3902	25.8	129
	10	0:15:43	8.8624	8.0	8.2100	15.0	8.1759	20.0	120
	11	0:15:51	7.5969	9.3	7.2707	17.4	7.1956	23.2	139
	12	0:14:0	7.4411	8.9	7.1915	16.5	7.1215	23.2	116
	13	0:16:21	7.4868	9.0	7.3081	17.0	7.0890	22.7	136
	14	0:8:18	5.9062	13.4	5.0051	21.4	4.8425	35.7	107
	15	0:7:4	4.0392	15.6	4.1553	27.3	4.8425	36.3	109
Aggregate Value			**6.4075**	**11.6**	**6.1414**	**21.8**	**6.0655**	**30.1**	**169.0**
Mavic Air 2	16	0:6:38	4.1252	14.6	4.0018	25.5	3.9858	34.0	102
	17	0:18:59	7.5923	9.2	7.0802	18.5	7.0994	23.7	166
	18	0:13:39	6.8513	10.7	6.6455	18.3	6.3794	25.6	128
	19	0:12:56	8.0372	8.5	7.8029	14.6	8.0008	20.4	102
	20	0:16:26	5.0192	14.0	4.9076	26.1	4.7372	34.8	209
	21	0:12:16	9.1065	9.1	8.1710	18.2	6.8722	27.3	109
	22	0:13:48	4.0079	15.1	3.9473	30.1	3.9381	42.2	211
Aggregate Value			**6.3914**	**11.6**	**6.0795**	**21.6**	**5.8590**	**30.0**	**146.7**
M300 RTK	23	0:17:1	8.7417	8.4	8.5434	14.0	8.7069	21.0	126
	24	0:13:18	6.2346	11.2	5.9937	19.1	6.0583	26.8	134
	25	0:22:29	13.134	7.1	11.045	13.5	10.549	16.9	135
	26	0:26:41	10.641	7.6	9.8213	13.5	9.6629	19.4	175
	27	0:25:51	8.9525	8.2	8.4705	15.2	8.3761	21.9	197
Aggregate Value			**9.5408**	**8.5**	**8.7748**	**15.1**	**8.6707**	**21.2**	**153.4**
Mavic 2	28	0:15:13	7.6689	8.4	8.1407	14.6	8.6594	23.4	117
	29	0:21:35	8.4391	9.4	7.1333	17.9	7.0158	28.1	197
	30	0:21:29	8.3854	9.5	7.8058	16.4	7.6738	25.7	180
Aggregate Value			**8.1644**	**9.1**	**7.6933**	**16.3**	**7.7830**	**25.7**	**164.7**
Mavic Mini	31	0:18:25	8.3382	10.6	6.6295	20.1	6.2721	30.2	181
	32	0:18:33	8.0766	9.9	6.7350	19.8	6.7538	25.4	178
Aggregate Value			**8.2074**	**10.3**	**6.6822**	**20.0**	**6.5128**	**27.8**	**179.5**
FPV	33	0:11:23	5.3383	12.1	5.3871	22.2	5.1617	33.3	133
Aggregate Value			**5.3383**	**12.1**	**5.3871**	**22.2**	**5.1617**	**33.3**	**133**



(a) (b)



(c)

Figure 2. Time block calculations for DFW flight data

The box plots in Figure 2 display the calculations gathered for the mean, median, and mode for DFW flights ranging between 6-12 minutes, 12-15 minutes, and 15-27 minutes. This allowed for a broader visualization at where the data stands regarding the duration of the flight time.

III.1.2. NIC, NAC based criteria for validating GPS integrity check using OpenSky Data

In this study, Automatic Dependent Surveillance-Broadcast (ADS-B) data from open sky network is used to detect dropouts. UND's team have designed an easy-to-use program that simplifies querying the Open Sky-Network data base, while also adding key performance metrics to the state vectors data which was previously in a separate table, and thus simplifying

the pre-processing steps to track GPS interference activities. We also designed an interface showcasing a Dashboard that points to "GPS interference hotspots" to serve as an additional safety measure for civilian aircrafts against known and unknown sources of GPS interference.

Open sky network is a non-profit, crowd sourced, off the shelf ADS-B receiver network that collects data from volunteers all over the world since 2013. This data is processed and stored in a central database. The database contains positional – Airborne and Surface, Identification, Velocity, operational status, and uncertainty metrics transmitted by aircrafts with ADS-B in the range of volunteer operated sensors.

The ADS-B is a device or unit on aircrafts that broadcasts aircraft state parameters at regular intervals without interrogation, which is an improvement on the previously used Mode S which required interrogation for message transmission.

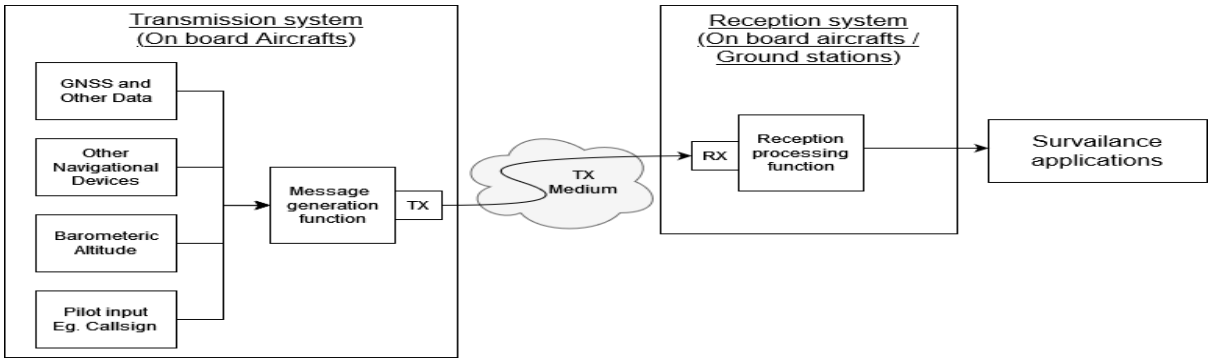


Figure 3. ADS-B System Overview

Automatic is a reference to the fact that the ADS-B transmits information without the need of operator intervention and Dependent indicates that the ADS-B depends on other air data systems like altimeters and GNSS Global Navigation Satellite Systems (GNSS) like the Global Positioning System (GPS) etc. to obtain the information that it transmits.

The information transmitted by the ADS-B are described in the Table 4.

Table 4. Types of Broadcast ADS-B

Message Type	**Information Transmitted**
Identification	Callsign, Wake Vortex Category
Airborne Position	Position, Altitude
Surface Position	Position, speed and track angle
Airborne Velocity	Vertical rate, GNSS and Baroaltitude difference, Ground Speed, Air Speed
Operational Status	Airborne status message, Surface status message, Capacity class, Operational mode, ADS-B version number, NIC supplement – A, Navigational accuracy category – position, Source integrity level, Horizontal reference direction, SIL supplement

Metrics to Track Uncertainty in Navigation Related Parameters

The ADS-B version 2 also broadcasts the following parameters (in addition to parameters listed above) as an indication of the accuracy or quality of the positional information being transmitted.

1. Navigation integrity category (NIC): NIC is an indicator of the accuracy the transmitted position. The higher the value of NIC, higher the position accuracy, and vice versa (Z. Liu et al. n.d.).
2. NIC replaced the Navigational Uncertainty Category (NUCp) parameter, which was used in Version 1 of ADS-B.
3. Navigation Accuracy Category (NAC): It is another metric that could be seen as a complementary indicator of NIC and can be used to determine the horizontal and vertical bounds of the position.
4. Surveillance Integrity Level (SIL): Probability estimation of measurements exceeding the containment radius.

We focused on NIC parameter, looking at how it is stored in the Open Sky Network database, and a method by which it is unified with state_vectors_data4 and queried together to integrate and produce data which combines state_vectors_data4 and the NIC parameter which resides in a separate table.

NIC: NIC is a clear indication of the accuracy of the position obtained by the on board GNSS system (Z. Liu et al. n.d.). The acceptable value of NIC is 7. Any values of 6 and below are indicators of abnormality (Federal Aviation Administration 2016). Thus, one can easily determine GPS interference activities by monitoring NIC. A loss in positional information for at least 10 seconds, followed by a drop in NIC (with 0 being complete loss) constitute a compromise in GPS Integrity (Z. Liu et al. n.d.).

GPS Jamming Exercises and Notices to Air Men (NOTAM):

There are timely exercises conducted by the military where the GPS signal in a pre-defined area and time. These exercises are a serious threat to civil aviation (IEEE Spectrum, n.d.). There have been numerous reports of aircrafts losing GPS connectivity due to these exercises.

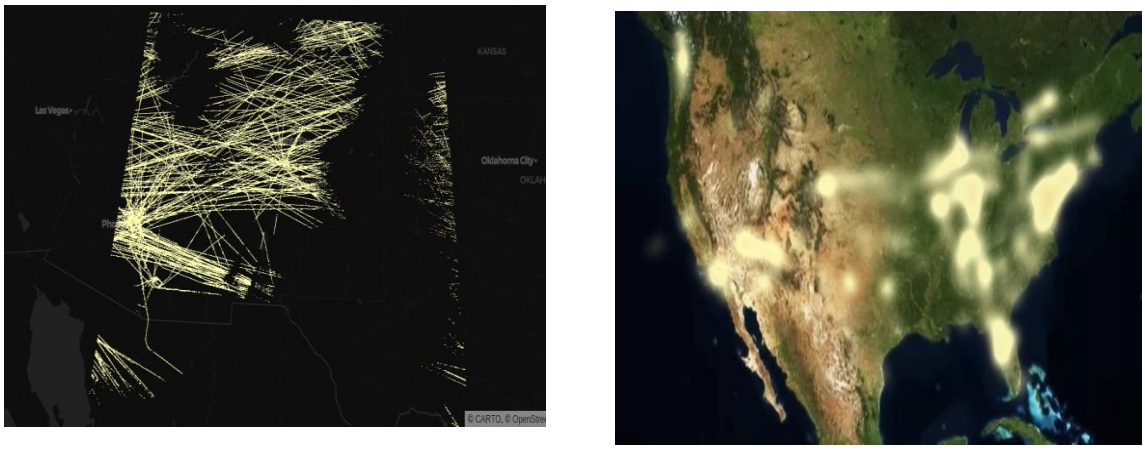
Notices are issued for hazards / change in facilities or conditions which is essential to personnel concerned with flight operations (Airport Authority of India 2022). These notices are also issued for GPS jamming events. The White Sands Missile Range seemed to be one of the hotspots for these exercises (estaff 2021; IEEE Spectrum, n.d.; Harris 2021).

NIC in OpenSky Database: The OpenSky database holds NIC in a separate table, in which the NIC is logged between timestamps (Min time and Max time) instead of a single timestamp as used the state vectors data. To get to drawing conclusions and analyze areas of interference, it would first be required to combine NIC with the state vectors data4. Our github Repository; open_sky (DECS Research 2022) does exactly this with a few bells and whistles. Given a query for the state_vectors_data4 table, it connects to the OpenSky IMPALA database, runs the query, saves the data obtained from the query to disk, once saved, it obtains the unique identifiers of aircrafts in the data obtained. With these unique aircrafts, it then queries the position_data4 table, obtains the NIC value, matches the timestamps, and creates files for each

aircraft, with the NIC and NAC value (where available) combined. It is also able to catch authentication time out errors, which if occurs, the query resumes from the last obtained call to rerun, thus automating the process of obtaining data from open sky which is a time-consuming task, thus freeing the user from waiting for queries to complete. On successfully obtaining the data from the remote database, it becomes easily available for analysis and processing.

The OpenSky database holds NIC in a separate table, in which the NIC is logged between timestamps (min. time and max. time) instead of a single timestamp as used the state vectors data. To get to drawing conclusions and analyze areas of interference, it would first be required to combine NIC with the state vectors data4. Our Github Repository; open_sky (DECS Research 2022) does exactly this with a few bells and whistles. Given a query for the state_vectors_data4 table, it connects to the OpenSky IMPALA database, runs the query, saves the data obtained from the query to disk, once saved, it obtains the unique identifiers of aircrafts in the data obtained. With these unique aircrafts, it then queries the position_data4 table, obtains the NIC value, matches the timestamps, and creates files for each aircraft, with the NIC and NAC value (where available) combined. It is also able to catch authentication time out errors, which if occurs, the query resumes from the last obtained call to rerun, thus automating the process of obtaining data from open sky which is a time-consuming task, thus freeing the user from waiting for queries to complete. On successfully obtaining the data from the remote database, it becomes easily available for analysis and processing.

Analysis of a GPS Jamming events. Now that the absence of NIC data from OpenSky was overcome and we had information on Jamming events, we tried to manually find an occurrence of GPS Interference in an area as described in NOTAMs. On analysis of a NOTAM on 19th April 2021, the area of interference was marked and queried from the OpenSky database. Fig. 4a shows an overview of all the aircrafts queried from the database at the time of the GPS interference described in the NOTAM.



(a) (b)

Figure 4. (a) Visual View of number of OpenSky Aircrafts Queried on 19th April 2021 (b) Reported Locations where NIC Level are less than 7.

On manual inspection of the data, we were able to find the previously mentioned pattern in aircrafts, and an example is show in Fig. 5. However, it is worthy to note that they were very scarce.

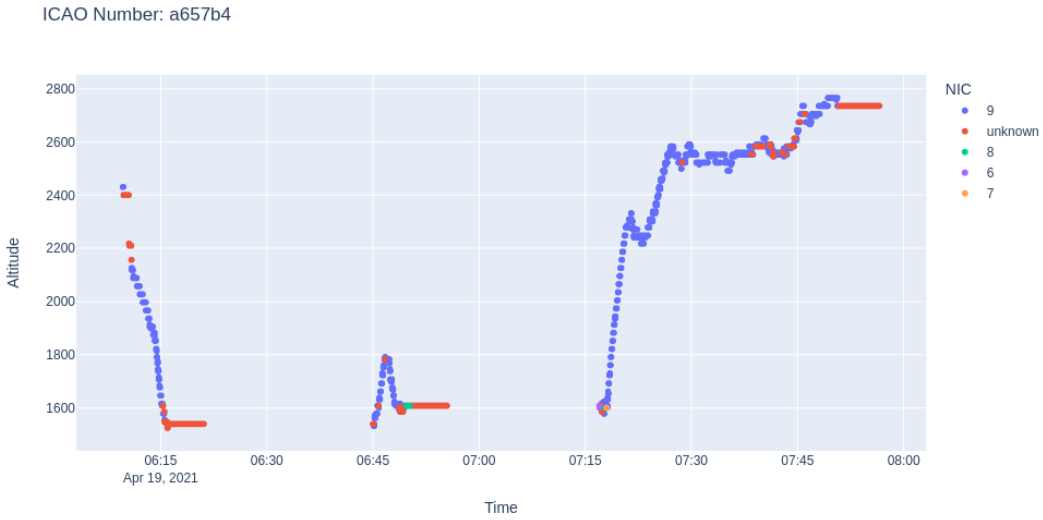


Figure 5. Possible GPS Interference.

Using NIC to Detect GPS Integrity Locational Hotspots. According to (Federal Aviation Administration 2016), a drop of NIC below 7 is considered abnormal. And now that we have a method to easily access NIC data along with the state vectors data, we will be looking at identifying GPS interference Hot Spots (areas of NIC drop below normal levels) using this data. We will create a dashboard that shows areas around the earth (that are in range of OpenSky receivers) where a drop in NIC has occurred, serving as an additional safety measure that can warn pilots about areas where unknown and known interference has occurred. Fig. 4b is an example of such a visual.

Additional criteria to explore further for ADS-B related anomalies:
Criteria – 1: Transmission Rates of ADSB: If there is no change in NIC/NAC/SIL, the ADSB transmits at a constant rate. However, if there is a change in these parameters, it would cause the ADSB to transmit at a higher rate (approximately 24 seconds as per (J. Sun, n.d.)). Detecting this increase in the transmission rate could serve as an additional criterion for any change in NIC/NAC/SIL.

Criteria – 2: Message Reporting Frequency. Filtering out messages from a single sensor and monitoring the number of messages received by that sensor for a given aircraft could help in detecting the increase in transmission rate. This could serve as an additional confirmation for change in key parameters. Simple statistical methods can be used to find a higher rate of transmission from normal transmission rates. The methods suggested by (Darabseh, Bitsikas, and Tedongmo 2019) highlighted a few drawbacks, one of which is the inability to use NAC as a parameter as it is not recorded for every single received message. The use of message count instead of NAC itself can help overcome this issue. (Murrian et al. 2020) have been able to use Low earth-orbiting satellites to detect GPS jamming incidents, we can use this data as a reference to check for these patterns in those areas.

Aircraft data was obtained from the OpenSky Network (Schäfer et al. 2014) using their API to pull 9 days of global data from their database with a myriad of features (see Fig. 6). These datasets were then parsed for unique ICAO24 aircraft tags which then had their data filtered and saved into individual files. There was difficulty distinguishing a dropout from an aircraft

making multiple trips due to ADS-B sensors being switched off for short lengths of time on the runway. This was mitigated by choosing a threshold of ≥15 minutes to separate the dataset into unique trips. Once these datasets were separated, the statistical analysis began. The columns of interest are the following: time, lat, lon, velocity, geoaltitude, and lastcontact. "lastcontact" is the column important to the calculation of dropouts.

An analysis of each flight using statistical methods allowed data to be scored and categorized as a dropout, noise, normal, or erroneous packet. Machine learning models were applied to predict these labels using only latitude, longitude, velocity, geo-altitude, and dropout length.

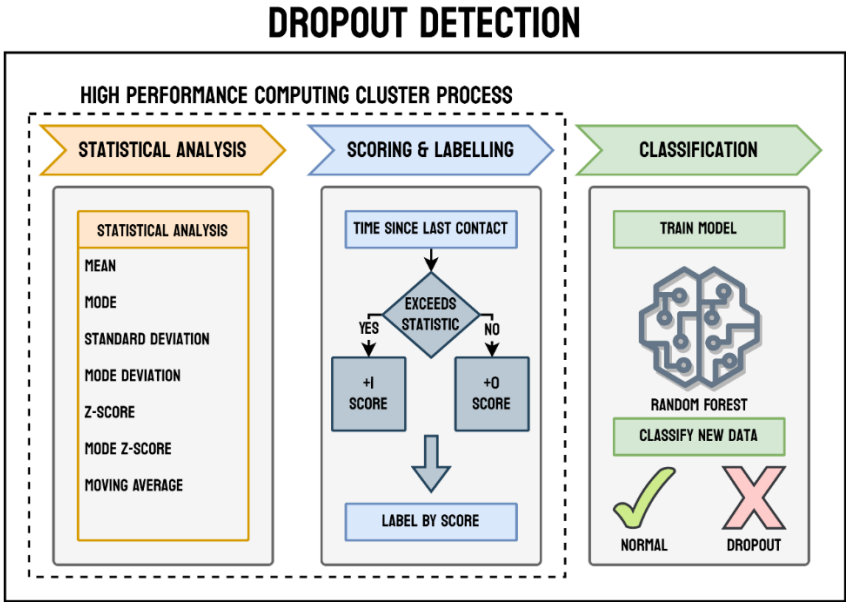


Figure 6. GPS/ADS-B Dropout Detection Framework.

Dropouts are defined as "larger than average time differences between communications" (Tabassum and Semke 2018b). A dropout is calculated by finding the difference between the current "lastcontact" and the previous.

Table 5. Batch data details

Batch	Unique Aircraft	Data Points	Dataset Type
1	55	691,764	Training
2	2	201,624	Testing
3	2	241,891	Testing

This is the basis for all calculations that follow, as it effectively measures the latency of ADS-B responses. ADS-B responses are supposed to come at a regular interval, normally every 2 seconds. Variations in dropout length indicate the extremity of the deviation from this interval. The mean, median, and mode of the dropout lengths were calculated, as well as the standard deviation and z-score of each point. The average dropout length was extremely skewed because of the extremity of some values, resulting in the difference between normal values and noise becoming unclear. To mitigate this, a method of calculating deviation from the mode was developed by replacing the average with mode in the standard deviation formula. This

eliminated the skew of the dataset and allowed the calculation of a modal z-score. Next, a simple moving average using a window of 25 was applied. Finally, a rolling signal-to-noise ratio was applied using methods similar to one researched for Wi-Fi offloading (Kumar and Gupta 2018). This can be used to find a threshold for separating dropouts from noise. The datasets were not injected with noise or modified in any way as they already contained a good number of points corresponding to each category. After statistics were calculated for each data point, they were scored based on how many values they were greater than. For example, if a dropout length exceeded three standard deviations it would get three points. Point values ranged from 0 to 10 and were based on average, mode, standard deviation, mode deviation, simple moving average, and signal-to-noise ratio. An examination of the points revealed a clear noise threshold score of 4 points. Points with a dropout length less than 0 were labeled as 'erroneous'. Scores of 1 were labeled 'normal', scores 2 to 4 were labeled 'noise', and scores greater than 4 were labeled as 'dropout'.

The data is now prepared to train the SciKit Learn models (Pedregosa FABIANPEDREGOSA et al. 2011; Buitinck et al. 2013). The input features for each model are the same: 'lat', 'lon', 'velocity', 'geoaltitude', and 'dropout _ length'. The output is the label of the data point: 'erroneous', 'normal', 'noise', or 'dropout'. Time series is excluded from the model on purpose so that it will not learn to detect past events that cause dropouts (weather, geomagnetic events, etc.) whilst training. Three machine learning models were attempted for the classification task: Random Forest Classifier, Support Vector Classifier with Linear Kernel, and K-Nearest Neighbor Clustering. Three "batches" of data were prepared as model inputs [Table 5]. Batch 1 was used for training as it is the most diverse. Each model was tested on Batches 2 and 3.

The Random Forest Classifier was trained using SciKit Learn's Random Forest Classifier model. An analysis of feature importance during testing showed 'dropout length' as the best predictor of labels, and removing it caused prediction accuracy to drop significantly. The use of different datasets was necessary to make sure the model was not overfitting. Random Forests generally do not overfit because it is an ensemble model of many different trees. The parameters were tuned using grid search cross validation with five folds (SciKit Learn's GridSearchCV), and then the best parameters were used for prediction. The best parameters were: {bootstrap = True, max depth = 80, max features = 'auto', min samples leaf = 5, min _ samples split = 8, n estimators = 100, random state = 42}. Efforts to prove that the model was overfitting yielded no evidence, so it can likely be generalized. Random Forest also boasts impressive performance, fitting the model in as little as 15 seconds.

The K-Nearest Neighbor model was very practical to optimize. The performance was great which allowed the number of neighbors to be varied and plotted in a reasonable amount of time. Analysis of the number of neighbors, 'k', from 1 to 30 2 showed a steep decline from 1 that began leveling out around 5. Lower values of 'k' correspond to a smaller window size which is essentially a measure of locality. The accuracy vs 'k' plot indicates that the model is more accurate when the locality is increased, which is not unusual for clustering. GridSearchCV was used to optimize hyperparameters, resulting in a single model with poor performance. The best parameters were algorithm=ball tree, n_neighbors=25.

Table 6. Comparison of model classification reports

Model	Metric	Batch							
		Batch 2				Batch 3			
RF	index	precision	recall	f1-score	support	precision	recall	f1-score	support
	dropout	0.396	0.466	0.428	1093	0.590	0.459	0.516	4029
	erroneous	0.986	1.000	0.993	17872	0.996	1.000	0.998	62434
	noise	0.963	0.957	0.960	93078	0.792	0.877	0.833	83828
	normal	0.964	0.965	0.965	89581	0.892	0.810	0.849	91600
	accuracy	0.962				0.877			
KNN	index	precision	recall	f1-score	support	precision	recall	f1-score	support
	dropout	1.000	0.014	0.027	1093	1.000	0.011	0.023	4029
	erroneous	0.451	0.211	0.288	17872	0.449	0.141	0.214	62434
	noise	0.493	0.483	0.488	93078	0.362	0.509	0.423	83828
	normal	0.463	0.527	0.493	89581	0.398	0.452	0.423	91600
	accuracy	0.476				0.384			
SVM	index	precision	recall	f1-score	support	precision	recall	f1-score	support
	dropout	0.922	0.218	0.352	1093	0.898	0.255	0.397	4029
	erroneous	0.988	0.623	0.764	17872	0.998	0.871	0.930	62434
	noise	0.490	1.000	0.657	93078	0.449	0.999	0.620	83828
	normal	0.000	0.000	0.000	89581	0.000	0.000	0.000	91600
	accuracy	0.518				0.575			

The Support Vector Classifier was tested on the same data and with multiple kernels. Initially linear, polynomial, sigmoid, and gaussian kernels were tested. The runtime of all kernels except linear was too poor to be practical, which is why linear was chosen. It very quickly became apparent that this model's performance was significantly worse than the other methods, with an average runtime of ~5 minutes. Iterating on this model was impractical because of this poor performance.

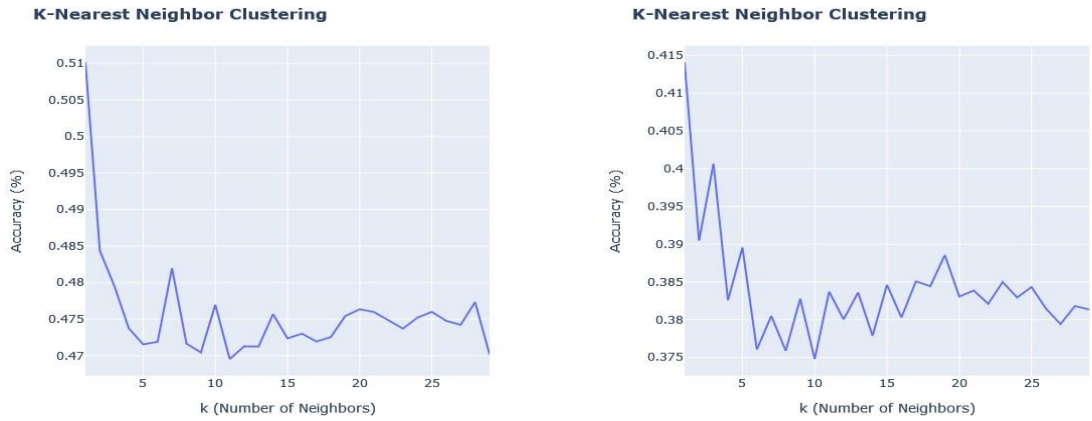
III.2.3. Results

Of the three models tested, Random Forest Classifier (RF) was the clear best. K-Nearest Neighbor Clustering (KNN) provided useful insight but was poor, and Support Vector Machine Classifier with Linear Kernel (SVM) was poor. The 'accuracy' and 'precision' metrics were used to analyze performance, but all metrics are shown in Table 6. The metrics precision, recall,

f1-score, and support are calculated from ratios of true positives (tp), false positives (fp), false negatives (fn), and number of occurences of each actual (not predicted) label in the test set. Precision is the ratio of true positives versus true positives plus false positives (Buitinck et al. 2013).

The Random Forest Classifier has impressive runtime and greater than 87% overall accuracy for both batches (see Table 6). It was able to correctly predict erroneous, noise, and normal data with 79% to 99% precision. However, dropout precision was only greater than 39%. These results are good, but dropout prediction is the core purpose of this project, and the model does not classify that category particularly well.

K-Nearest Neighbor Clustering performed poorly for both test sets. The plot of accuracy vs number of neighbors (k) indicates high locality for the data points, which may mean that the data is highly dependent on its closest neighbors (see Fig. 7). Larger numbers of neighbors could be causing the clusters to ignore minute details and misclassify data. The optimized KNN model had greater than 38% accuracy overall, but the precision for all categories except dropouts was between 36% to 49%. The dropout precision was 100%, likely due to misclassification.



(a) (b)

Figure 7. KNN Accuracy vs Number of Neighbors (k) for (a) Batch 2 and (b) Batch 3.

The Linear Support Vector Classifier was very poor and provided little useful information. The accuracy was greater than 51% for both test sets (see Table 6). The precision for dropouts and erroneous was 89% to 98% and noise was around 49%. ”Normal” labels were never predicted by the model resulting in mass misclassification. The runtime performance was very slow for linear kernel (~5 minutes) even with the Intel® Extension for Scikit-learn which improves performance. The polynomial, gaussian, and sigmoid kernels have time complexities that can reach $O(n^2)$ or $O(n^3)$ in some cases (Bottou and Lin, n.d.; Simon and List 2009), which is probably why training never finished for Batch 1 with ~700,000 data points. The accuracy and f1-score for the linear kernel were probably so poor because the features are too complex to classify well with a straight line through a hyperplane.

III.2.4. Conclusion

The Random Forest Classifier is easily the best model in terms of classification accuracy and performance but struggles to classify dropouts. K-Nearest Neighbor Clustering is not a good fit for this problem but was able to provide insight into the locality of dropouts. The Linear Support Vector Classifier is not a good fit for this problem because the linear kernel is too simple to achieve accurate results and the dataset is too large for use with polynomial, gaussian, or sigmoid kernels.

III.2.5. Problems and Future Work

Deleting duplicate data received from multiple sensors at the same time is a temporary fix and should be investigated further. Separating trips by finding time differences ≥15 minutes was decided by a judgement call and needs revisited. Labelling data based on statistics is unreliable because the categories are not well-defined. The score thresholds were decided based on judgement calls after analysing plots. The 'erroneous' category does not encompass all causes of erroneous data and should be expanded. More training data is needed to improve the Random Forest Classifier's dropout prediction accuracy. Future work will include researching duplicate data causes, better trip splitting, refining label definitions, and Random Forest Classifier improvement. The K-Nearest Neighbor Clustering and Linear Support Vector Classifier models will not be the focus of future work but may be revisited if a future use is found for them. The Random Forest Classifier will be improved by training it on more batches of data as well as a direct approach to hyperparameter optimization.

III.3. Imputation of GPS/ADS-B Dropout using OpenSky Network

The goal of this section of the project is to investigate the ADS-B metadata from an open-source sensor network and use a machine learning framework to impute missing data points for GPS/ADS-B such as position, direction, etc. This section first reviews related work for imputation and then presents the dataset collection and features. Next, it covers the machine learning methods that will be used for this project. Lastly, the experimental section where data analysis was carried out and the corresponding results are presented.

The data from the OpenSky Network were downloaded using a Python script. The information was arranged by day and hour in each parquet file. These files contain all the aircraft's abstracted observations for that hour. The main components of the dataset are timestamps, positions (i.e., latitude, longitude, and altitude), velocity, and heading. This study was taken from February 19, 2022, to February 27, 2022, with six flights totaling approximately 60,100 points. Table 7 lists the characteristics of the OpenSky dataset.

Table 7. Features of the OpenSky dataset

Field Name	Field Purpose	Sample Data
time	the Unix (epoch) timestamp for which the state vector was valid. Each aircraft was active within the coverage of OpenSky ADS-B Receivers at that second.	1479957078

icao24	the 24-bit ICAO transponder ID can be used to track specific airframes over different flights. This ID should never change during a registration period of an airframe, which doesn't change very often.	780db8
lat	last known latitude of the aircraft in decimal degrees.	118.59931
lon	last known longitude of the aircraft in decimal degrees.	22.916793
geoaltitude	the actual height of aircraft above sea level in meters.	8839.2

Pre-processing data aims to transform raw data into a more usable and effective format for subsequent processing steps. Each flight was broken down into individual trips. For an aircraft, split points were defined as two consecutive points in time designated as the last point of one track and the first point of another track. The tracks were split into 15-minute intervals, assuming that the flight time had passed since the last contact and tagged with numeric numbering with "trackId" and saved as a separate parquet file. These data files will be used to train machine learning models.

The overview of the Machine Learning Framework for Imputing ADS-B/GPS Dropout Data is shown in Fig. 8. pySpark, a Python API for Apache Spark, was used to process the feature columns (an open-source distributed computing framework and set of libraries for real-time, large-scale data processing). The percentage of missing values (the "missing rate") is then introduced to these data at 10%, 20%, and 30% at random and continuous intervals. The missing values are then imputed using machine learning algorithms. After the values are derived using machine learning algorithms, they are forwarded to the validation section, which evaluates the performance of each machine learning algorithm using MAE and RMSE metrics.

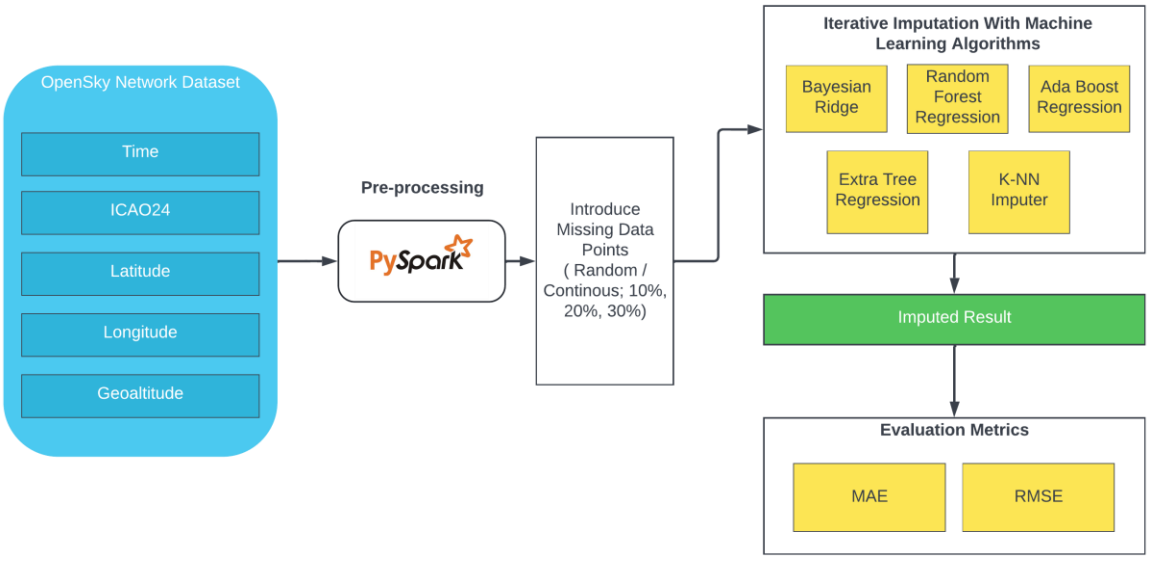


Figure 8. Overview of machine learning framework for imputing ADS-B/GPS dropout data.

The Bayesian Ridge, Random Forest, Adaptive Boosting, Extra-Tree Regressor, and k-Nearest Neighbor regression methods were investigated in this study.

(i) The Bayesian Ridge technique is a regression model with a regularization parameter. It's an estimator that uses a calculation with an additional regularization term equal to the sum of the values' squares to assume and predict the target.
(ii) Random Forest (RF) algorithm is used for classification, and regression tasks. This is a machine learning algorithm that creates the number of trees during the training period and then provides the individual trees' output class. The RF algorithm deals with various types of missing data. When the tree grows, the imputation is done adaptively, and all missing values are replaced at the end of each iteration.
(iii) The Adaboost algorithm, also known as adaptive boosting, loops a weak classifier. The weight of objects for classification is redistributed after each call and the incorrectly classified objects increase with each iteration, and the new classifier focuses on these objects. Calculating average weighted classifiers is used to make predictions. As the method continues to attempt to correct the data's incorrect classifications.
(iv) Extra Tree Classifier is a modified version of bagging classifiers. It uses standard tree techniques, but with the added goal of increasing efficiency and accuracy. The difference between other tree-based algorithms is the split of the node as they are randomly selected cut points and building trees using total learning samples.
(v) k-Nearest Neighbor (k-NN) Imputation is another technique for imputing a missing score. It first finds the most similar records in the dataset using the Euclidean distance. The technique uses the mean value over the nearest neighbors. This method performs well on datasets having a strong local correlation structure. However, this method is computationally expensive for large datasets, because finding the most suitable -nearest neighbors is based on searching the whole dataset.

The experimental settings are listed in Table 8. Each experiment is executed five times, with different missing rates of 10%, 20%, and 30% (randomly and continuously), and evaluated using two metrics: mean absolute error (MAE) and root mean square error (RMSE). The MAE and RMSE are defined in Equations (3) and (4) respectively.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|x_i - \hat{x_i}| \qquad (3)$$

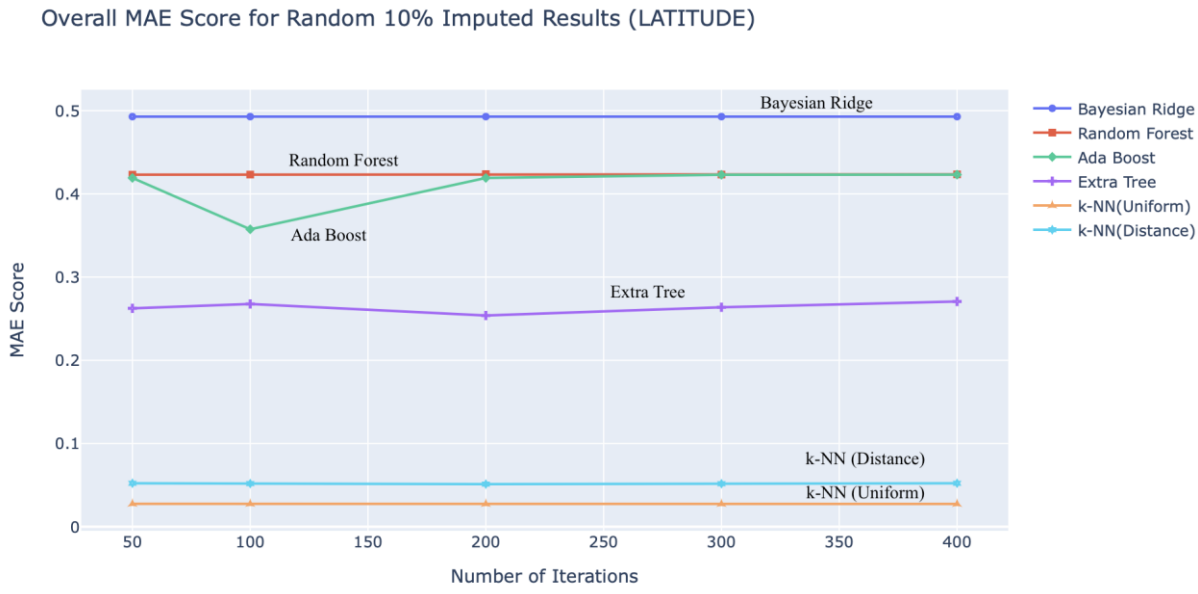
$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}|x_i - \hat{x_i}|} \qquad (4)$$

Table 8. Overview of the experimental settings.

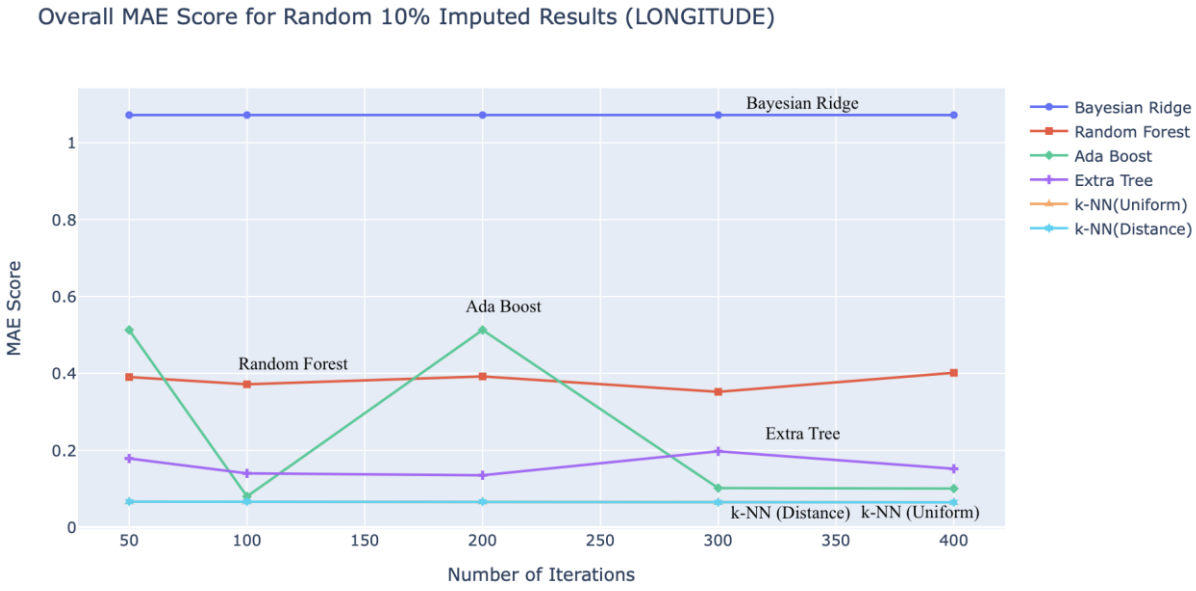
Imputation Methods	**Hyperparameter Settings**	
	Name	**Values**
BRR	-	-
RF, ABR, ETR	n_estimators	(10, 50, 100)
k-NN	n_neighbors weights	(1, 2, 3, 4, 5, 10, 20, 30, 50) (uniform, distance)

Note: All the above Imputation methods ran with multiple iterations - 50, 100, 200, 300, and 400 times to compute the results with missing ratios of 10%, 20%, 30% (RANDOMLY and CONTINUOUSLY)

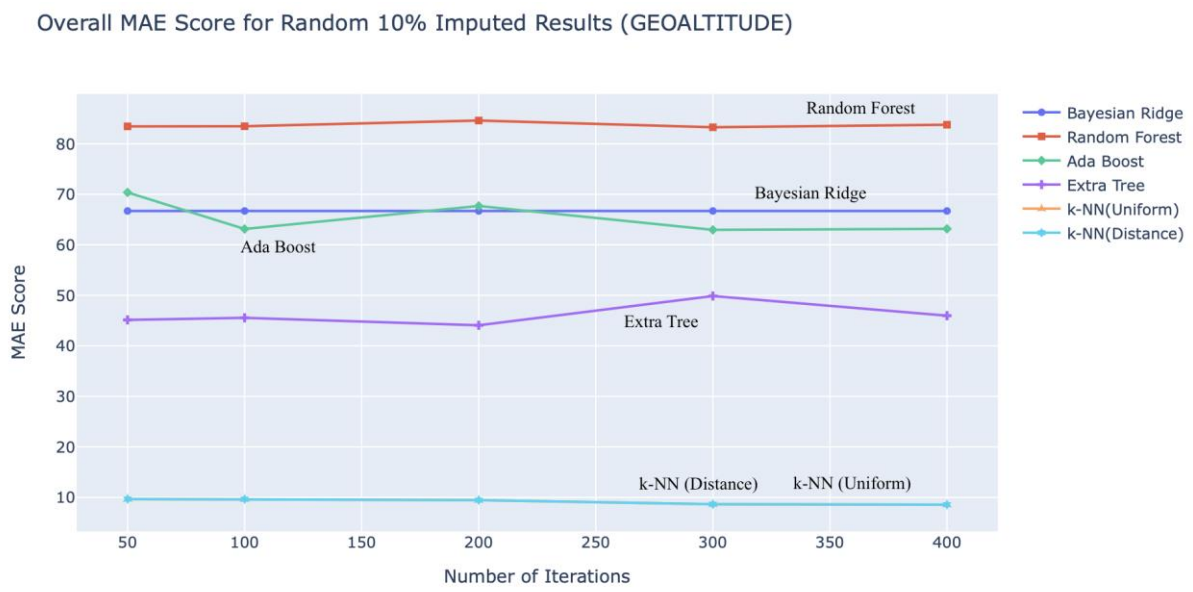
This study described a machine-learning approach to impute flight data. Our experiments demonstrated different percentages of missing rates like 10%, 20%, and 30% randomly and continuously for five different machine learning methods such as Bayesian Ridge, Random Forest, Adaboost, Extra Tree, and k-NN. Results show that k-NN performed better compared to other machine learning models to impute the parameters such as latitude, longitude, and geoaltitude. Fig. 9 and Fig. 10 shows the MAE and RMSE Score of 10% Imputed Results for 10% random missing rate of data points and the overall score for 10%, 20% and 30% random missing rates score has been shown in Table 9.



(a)

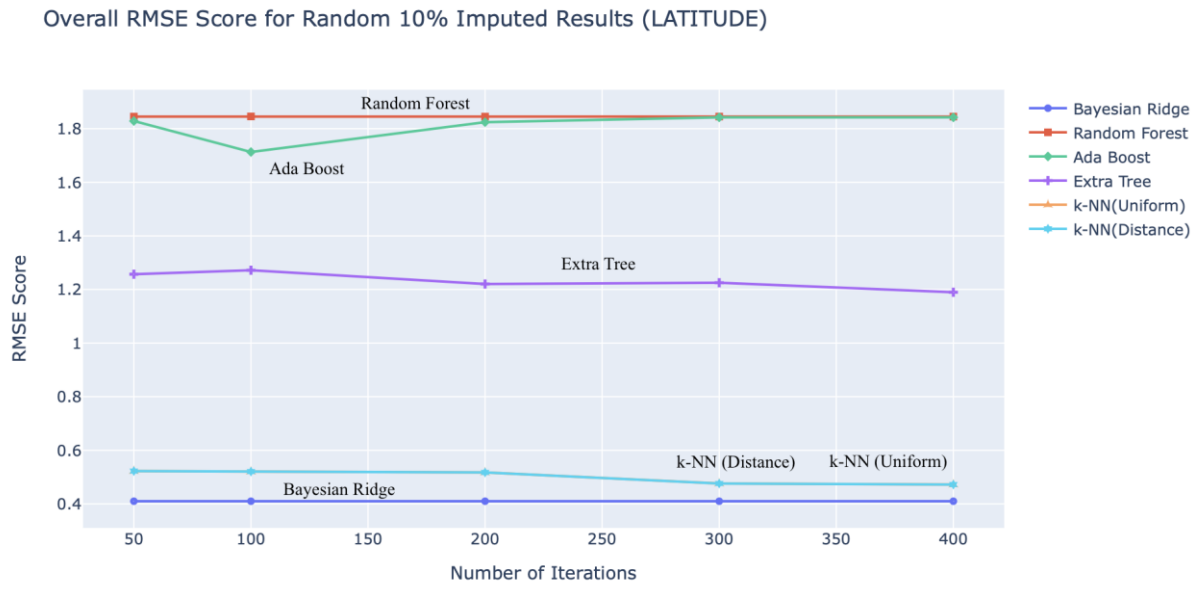


(b)

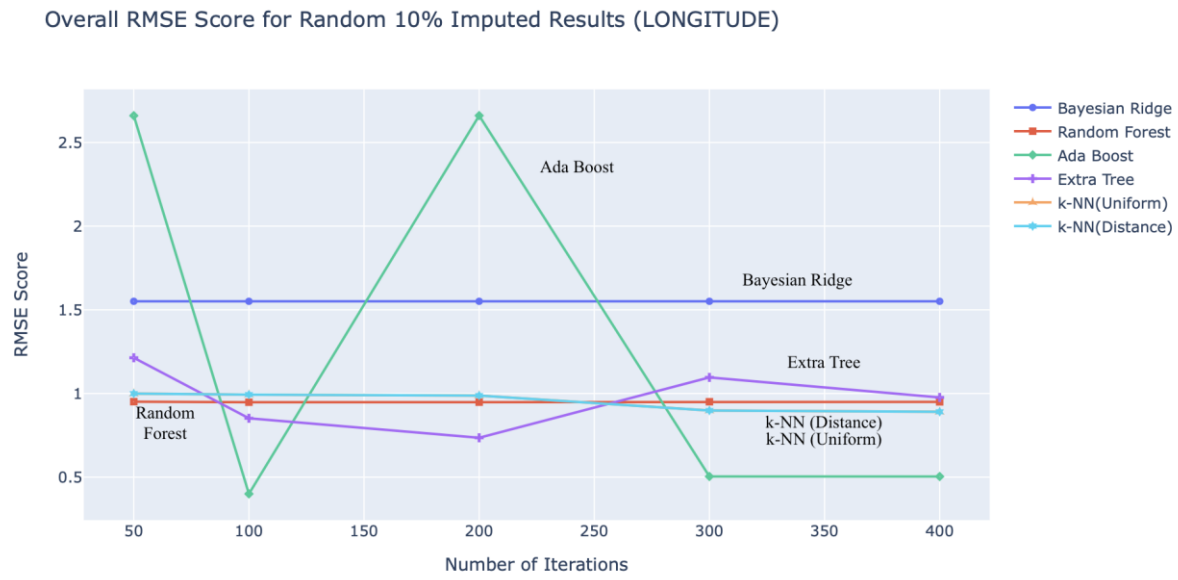


(c)

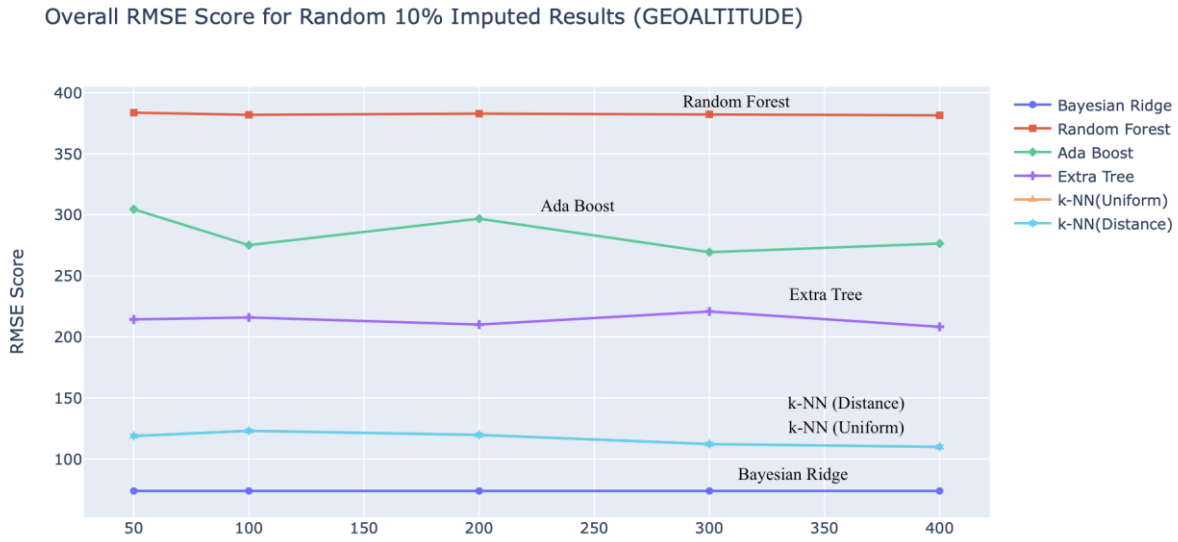
Figure 9. Comparison of MAE score for (a) latitude – 10% randomly imputed, (b) longitude – 10% randomly imputed, and (c) geoaltitude – 10% randomly imputed.



(a)



(b)



(c)

Figure 10. Comparison of RMSE score for (a) latitude – 10% randomly imputed, (b) longitude – 10% randomly imputed, and (c) geoaltitude – 10% randomly imputed.

Table 9. Overall MAE and RMSE Score for Random 10%, 20% and 30% Imputed Results for different Machine Learning Models

	Iterations	**Latitude**	**Longitude**	**Geoaltitude**	**Latitude**	**Longitude**	**Geoaltitude**
		MAE	**MAE**	**MAE**	**RMSE**	**RMSE**	**RMSE**
10%		**Bayesian Ridge**					
	50	0.492830079	1.072055268	66.69919011	0.40998854	1.550869284	73.80599744
	100	0.492830079	1.072055268	66.69919011	0.40998854	1.550869284	73.80599744
	200	0.492830079	1.072055268	66.69919011	0.40998854	1.550869284	73.80599744

	300	0.492830079	1.072055268	66.69919011	0.40998854	1.550869284	73.80599744
	400	0.492830079	1.072055268	66.69919011	0.40998854	1.550869284	73.80599744
		Random Forest					
	50	0.42503186	0.207037769	85.79975775	1.845060287	0.951028196	383.5995599
	100	0.425272302	0.205772608	84.98609983	1.845658094	0.947752522	381.8742073
	200	0.425025621	0.205588445	85.37631901	1.845061456	0.948046729	382.8347347
	300	0.425039307	0.206982654	85.59228839	1.845123707	0.9505982	382.2870004
	400	0.424999395	0.206152139	84.76511212	1.845013614	0.949825357	381.4528691
		Ada Boost Regression					
	50	0.419165228	0.512959996	70.36778214	1.828666381	2.660371428	304.5048345
	100	0.357435959	0.080153501	63.13325494	1.713001257	0.400138272	275.1160451
	200	0.419147449	0.513065477	67.68864518	1.824295366	2.660755977	296.8066023
	300	0.42296628	0.10185721	62.96341611	1.842142721	0.504132463	269.3876638
	400	0.423140092	0.100634107	63.1645789	1.84217706	0.503919208	276.4536648
		Extra Tree Regressor					
	50	0.262449672	0.178762603	45.13388782	1.256953514	1.213501815	214.3165176
	100	0.2676982	0.140112999	45.54510798	1.271853629	0.851870076	215.888596
	200	0.25384151	0.135195912	44.06861921	1.22039445	0.735435379	210.0778027
	300	0.263733453	0.197530451	49.86125902	1.225211218	1.096061922	220.7851158
	400	0.270658265	0.152120967	45.97270213	1.189685226	0.9764261	208.2632972
		K-NN (Uniform)					
	1	0.027532166	0.066613315	9.662930198	0.522493125	0.999307847	118.8085095
	2	0.027487276	0.066568082	9.641451088	0.520611007	0.993298562	123.0517479
	3	0.027419963	0.066353669	9.455552998	0.517350793	0.987595268	119.6967444
	4	0.027383505	0.064974578	8.651886915	0.476222426	0.898431802	112.2230434
	5	0.027399683	0.064794468	8.562564323	0.472245738	0.890847045	109.9124322
	10	0.027428692	0.065292114	8.464688216	0.455827012	0.86521706	104.773125
	20	0.027161867	0.066208489	8.398535101	0.421477021	0.81622559	97.32200922
	30	0.027063793	0.065744485	8.312910983	0.400315326	0.773790004	91.99387661
	50	0.02743398	0.06683715	8.498259174	0.3881106	0.737222914	88.03382786
		K-NN (Distance)					
	1	0.027532166	0.066613315	9.662930198	0.522493125	0.999307847	118.8085095
	2	0.027484898	0.066566884	9.640831459	0.520610997	0.99329856	123.0526078
	3	0.027416004	0.066351613	9.457178016	0.517350773	0.987595263	119.6973065

	4	0.027350133	0.064931037	8.646724263	0.475982231	0.898158033	112.1622027
	5	0.027368772	0.064759	8.559351435	0.472113297	0.890699001	109.8792518
	10	0.027403484	0.065253267	8.471833034	0.456803176	0.865817011	104.8702989
	20	0.027172885	0.06628078	8.455892749	0.429489914	0.828011282	99.48241624
	30	0.027038094	0.065566315	8.344993921	0.416535875	0.796425152	96.07001966
	50	0.027027155	0.066206696	8.404728221	0.407426537	0.774048722	93.69566042
20%		**Bayesian Ridge**					
	50	1.001440618	2.143181566	136.1931786	2.531412876	5.867462823	397.2331747
	100	1.001440618	2.143181566	136.1931786	2.531412876	5.867462823	397.2331747
	200	1.001440618	2.143181566	136.1931786	2.531412876	5.867462823	397.2331747
	300	1.001440618	2.143181566	136.1931786	2.531412876	5.867462823	397.2331747
	400	1.001440618	2.143181566	136.1931786	2.531412876	5.867462823	397.2331747
		Random Forest					
	50	0.853284114	0.415601724	168.6988073	2.596711308	1.352138522	530.0751282
	100	0.853561791	0.536176789	168.8326134	2.597923895	1.803540036	528.0109163
	200	0.853439108	0.56689016	168.6252493	2.597926228	1.952584616	529.8238888
	300	0.853580378	0.473358532	168.6626488	2.598405781	1.576363706	527.2290532
	400	0.853707719	0.545762927	169.2392705	2.598183749	1.859514906	529.0909613
		Ada Boost Regression					
	50	0.871732081	0.203017736	117.4987618	2.655199092	0.711810066	361.2073629
	100	0.863576912	1.082943185	136.9195778	2.635761525	3.889943035	423.5410716
	200	0.863443755	0.202337852	121.6230968	2.634549672	0.711218723	374.0147052
	300	0.871807053	1.083035116	140.5211972	2.65513633	3.889613126	439.0857848
	400	0.863088226	1.082146455	142.5492147	2.628829265	3.88879694	447.3231389
		Extra Tree Regressor					
	50	0.524985546	0.386929886	101.2159513	1.645688844	1.486674822	324.112877
	100	0.543263843	0.364148455	89.65644609	1.818989312	1.605462529	308.0580219
	200	0.555658527	0.322224604	99.11257941	1.799297804	1.388422627	327.7662453
	300	0.533402841	0.31483182	97.09117162	1.706439102	1.347140134	313.0170087
	400	0.526922365	0.353347737	94.23093472	1.785606704	1.556056105	319.0003721
		K-NN (Uniform)					
	1	0.052211182	0.127278725	18.99541256	0.719008661	1.381194445	178.0572183
	2	0.051842919	0.126288816	18.57815862	0.710195059	1.366574837	176.6601792
	3	0.051185014	0.125186043	18.16247219	0.70086535	1.35335673	174.9776793

	4	0.05168486	0.122932079	16.22559244	0.658401553	1.251413762	164.3371914
	5	0.052217724	0.123335397	16.23879129	0.658822834	1.242233206	163.7276441
	10	0.05292779	0.124996161	16.25345809	0.636590946	1.201081894	159.2767255
	20	0.052253532	0.124361673	15.95882572	0.586177933	1.114839486	148.8045905
	30	0.053433	0.126237675	16.24964947	0.560844059	1.056164844	143.1783207
	50	0.053388274	0.127037339	16.57420856	0.537819862	1.006760928	138.7271754
		K-NN (Distance)					
	1	0.052211182	0.127278725	18.99541256	0.719008661	1.381194445	178.0572183
	2	0.051836202	0.126283592	18.58254421	0.71019495	1.366574789	176.6668969
	3	0.051117866	0.125096168	18.15650435	0.700669255	1.353138856	174.9372343
	4	0.051532657	0.122719808	16.199979	0.658018854	1.250981768	164.2515226
	5	0.052031155	0.123073788	16.20697591	0.658201012	1.241554636	163.5994479
	10	0.052746927	0.124678887	16.23010066	0.635550864	1.199813024	159.0842087
	20	0.052093002	0.123885512	15.93096025	0.599952129	1.134303066	151.6275712
	30	0.052410167	0.124219034	15.94515483	0.586250379	1.093532826	147.9187539
	50	0.052514625	0.125333568	16.0520285	0.576108152	1.068014177	145.4699985
30%		**Bayesian Ridge**					
	50	1.481995801	2.143181566	136.1931786	3.072484836	5.867462823	397.2331747
	100	1.481995801	2.143181566	136.1931786	3.072484836	5.867462823	397.2331747
	200	1.481995801	2.143181566	136.1931786	3.072484836	5.867462823	397.2331747
	300	1.481995801	2.143181566	136.1931786	3.072484836	5.867462823	397.2331747
	400	1.481995801	2.143181566	136.1931786	3.072484836	5.867462823	397.2331747
		Random Forest					
	50	1.26599161	0.777679427	246.8213853	3.15871517	2.151855922	630.4123546
	100	1.265159075	0.891332245	247.9101044	3.154628379	2.537227123	634.3379042
	200	1.265532773	0.790574805	247.5010923	3.158079278	2.202231468	630.5059797
	300	1.265471489	0.684391522	245.309928	3.153226028	1.859195283	627.8685357
	400	1.26454382	0.849562411	247.5201958	3.155991989	2.394856967	630.8843684
		Ada Boost Regression					
	50	1.08121137	1.525872855	195.6110509	2.968547826	4.678705472	505.4460438
	100	1.275760996	1.588925663	193.8276628	3.18232925	4.713468159	489.722142
	200	1.26389592	1.588496314	210.1241655	3.152522516	4.711412569	531.0752002
	300	1.076725248	1.526513568	199.7008029	2.961372944	4.68209517	505.7816625
	400	1.079563902	1.5266325	194.4847214	2.956299362	4.682976969	488.7769995

		Extra Tree Regressor					
	50	0.822731562	0.434137247	144.2886901	2.140863437	1.57749114	371.4550778
	100	0.790392108	0.567727916	137.1704646	2.105545942	1.9165919	358.4916387
	200	0.751915396	0.512453231	130.3635017	2.059046699	2.120848806	359.2821502
	300	0.817057063	0.494198593	140.4390633	2.207896161	1.779435214	384.7146917
	400	0.796255525	0.594635932	152.2765229	2.238883754	1.832526833	404.6523813
		K-NN (Uniform)					
	1	0.083244152	0.199689815	29.54889967	0.909392718	1.729831857	218.5401873
	2	0.083442615	0.198942091	28.92848386	0.902454397	1.706595759	210.8261937
	3	0.083633938	0.198581052	28.34331582	0.897467745	1.695498829	206.4820601
	4	0.088766893	0.203467828	26.63933825	0.863811621	1.592937046	195.4626876
	5	0.08878445	0.203383042	26.55753335	0.856483298	1.577391569	193.0639326
	10	0.089752713	0.204640682	26.28673135	0.821027034	1.501998923	181.4859477
	20	0.089077668	0.204994889	26.023339	0.754367911	1.399694222	167.9504325
	30	0.08890798	0.205905968	26.17631093	0.717047994	1.329433915	159.1185822
	50	0.088564849	0.206263468	26.68409487	0.694794175	1.279744994	153.4732661
		K-NN (Distance)					
	1	0.083244152	0.199689815	29.54889967	0.909392718	1.729831857	218.5401873
	2	0.083510275	0.19905265	28.9434169	0.903136178	1.707372206	210.9957507
	3	0.083636973	0.198605527	28.34929065	0.897745681	1.695816912	206.5570702
	4	0.08866672	0.203347662	26.62579183	0.863974692	1.593128587	195.5080095
	5	0.088635996	0.203190576	26.54187277	0.856481442	1.577361464	193.0511823
	10	0.089058851	0.20363571	26.26429506	0.821477257	1.503713989	182.447644
	20	0.088283274	0.203061797	25.9146774	0.779312391	1.432432211	173.5492318
	30	0.088531605	0.20444206	26.02757937	0.763150203	1.395277302	169.3763042
	50	0.088372588	0.204886743	26.06445309	0.755763824	1.374126643	167.1227356

In future, other data fields, such as velocity and heading, will be explored. In addition, deep learning approaches such as GAIN (Generative Adversarial Imputation Networks), MIDAS (Multiple Imputation with Denoising Autoencoders), and LSTM (Long Term Short Memory) imputation approaches will also be extended.

IV. GPS and ADS-B Signal Jamming Potential Mitigation Assessment

Jamming is the process of interfering and blocking radio communications using frequency transmitting devices at the same working frequency as the target device. The jamming transmission introduces interference noise to the target signal which can introduce inaccuracies or cause the signal to dropout entirely. GPS and ADS-B functionality is based on RF transmission, making UAV operations vulnerable to jamming effects. (Yu 2012) (Leonardi and Piracci 2018). For the ADS-B signal Jamming Risk Classes, jamming is defined as the intentional and illegal process of interfering and blocking radio communications using frequency transmitting devices at the same working frequency as the target device. A jamming intervention may introduce noise to the main signal which can introduce inaccuracies, or even block and replace the desired data with the jamming signal.(Yu 2012) (Leonardi and Piracci 2018). GPS jamming methods are low-cost and increasingly accessible to the general public, introducing increased potential for jamming occurrences to impact GPS-informed navigation in UAV operations. For instance, UAV operations within urban areas take place in concentrated RF environments with high levels of noise, degraded signals, signal reflection, and other RF issues, disrupting operations relying on location and position determination using GPS signaling, and therefore impacting ADS-B effectiveness.

GPS and/or ADS-B jamming events that impact one or more drone operations in the NAS is expected to be a frequent event. Four mitigation schemes have been identified and assessed. The four schemes evaluated are optical flow, geomagnetic navigation, cellular signal navigation, and Wi-Fi navigation.

IV.1. Mitigation Strategy: OPTICAL FLOW

Optical flow techniques are solutions based on natural behaviors observed by insects and birds. These algorithms analyze pixel motion between two two-dimensional images as a projection of the three-dimensional motion of the objects relative to the visual sensor (Chao 2013. The navigation information obtainable through optical flow fields includes rotational and transnational velocities along terrain information expressed in body frame. Moreover, four methods are commonly used for navigation purposes corresponding to differential methods, region-based matching processes, phased-based methods and fusion based methods. For navigation purposes, the most common method is the Lucas and Kanade approach, which calculates the velocity of features found and tracked over two consecutive images. This method assumes slight changes between the images, constant brightness, and smooth spatial motion (Fontani 2014). Before obtaining a vector field, this algorithm requires reliable features in the images for further tracking using feature detection algorithms as the Shi-Tomasi, Features from Accelerated Segment (FAST), Scale Invariant Feature Transform (SIFT) and Speeded-up Robust Features (SURF) corners.

There exist multiple approaches to determine the optical flow from video sequences based on early image processing. Most of them are governed by the assumptions of constant brightness, small motions, and spatial smoothness. Along with the advances of efficient optical flow procedures, new applications have seen the inclusion of such methods as navigation of fixed wings and unmanned vehicle systems. Consequently, sensor hardware and reference motion

models are sections that require emphasis in optical flow developments for vehicle navigation purposes.

Motion Models based on optical flow are defined as projections of 3D relative motions into the observed 2D plane from the camera. This motion field should match the apparent motion when the assumptions are held, which corresponds to the optical flow (Zhang 2016). The motion fields obtained by the motion models serve as a reference for optical flow obtained by the computer vision algorithms. Here, important information such as the angular and translational velocities can be obtained or estimated for navigation. Two major approaches for deriving ideal motion field estimation are presented and denotes as the pin-hole image plane approach and spherical imaging surface, which are models based on biological eyes and compound eyes respectively.

Finally, by integrating ideal camera motion models along with algorithms of feature detection and tracking for optical flow calculations, Bayesian estimations may be integrated to obtain better estimations of the velocity by including inertial information (Sum 2013). This combination allows to eliminate noise due to images imperfections and outlawyers in feature selections. Among these Bayesian methods, a regular commonly used approach is the Kalman Filter, which integrates optical flow calculation into the ideal model and data provided by gyroscopes and accelerometers.

Cost – Rank 2

The cost implementation of this type of mitigation strategy can vary between a cost ranking of 3-4, based on the selection of camera and processing algorithm to be selected to compute the optical flow. Low-cost platforms can run optical flow under the cost of $250 with acceptable performance since the most common implementation does not require high-end cameras and complex algorithms. A clear example of a low-cost UAV capable of running optical flow satisfactory is the Mambo Parrot drone with a total cost of $150. For more advanced applications, the cost can increase based on the level of accuracy and mission requirements.

Technical Readiness - Rank 4

Optical flow has a ranking 4 since some systems are commercially available and is a common feature included in commercial high-end drones from DJI, Skydio, Sony among other brands. It provides advantages not only in pose estimation but also obstacle avoidance and path planning. The Pixhawk flight controller, one of the most used platforms for UAV research and development, offers for example as a peripheral a camera for an included optical flow algorithm.

Ease of Implementation/Use – Rank 3

Optical flow can be categorized as 3 with moderate modifications. Its implementation could be relatively easy based on the platform selected. For custom development often public libraries for algorithms selection are available as for example OpenCV which requires minor to moderate modifications depending on the final scope of implementation. Other platforms as Ardupilot and Pixhawk require minor modification for implementations of this system and Commercial drones do not require any manipulation at all from the final user in terms of algorithm manipulation or hardware integrations.

Size, Weight, and Power (SWaP) - Rank 3

This parameter can be adjustable based on the vehicle requirements since a wide variety of combinations of cameras and computer boards can be performed, selections from different sizes, weights and power consumptions characteristics.

The camera size is small in general as for example the CelePixel, CeleX5-MP or the PX4Flow, although It can vary from one inch by one-inch chipsets to boxes of couple inches as for example the DVXplorer and DAVIS346, which may include additional utilities or even microcontrollers. Additionally, to the camera, often is required a microcontroller that runs the algorithm if the camera does not provide. This element can be often the main Flight controller or additional companion bords of average sizes of 3 to 4 inches which are small for the average UAV dimension. Industrial UAVS may use bigger elements proportional to the size of the vehicle. Currently commercial vision cameras consume less than 800mW in total.

Impact– Rank 4

The implementation of Optical flow has provided significant improvement in sensing and navigation capabilities for UAVs. And it's a feature that can be found in commercial UAVS with high autonomous capabilities. Notwithstanding, this mitigation strategy still has its own set of limitations and weaknesses as for example its application over areas with even characteristics as constant colors or patterns, where the algorithm cannot perform a strong feature detection and therefore provides erroneous or null data.

Effectiveness Score – 16 High

IV.2. Mitigation Strategy: GEOMAGNETIC NAVIGATION

One alternative proposed even before the GPS era is the terrain navigation technique based on geomagenetic contours. Back in 1940, Goodyear Aircraft corp. Started developing the Automatic Terrain Recognition and Navigation System (ATRAN), a radar-map matching system capable of correcting the flight path deviation by correlating measurements from a radar scanning antenna with a series of maps on board a missile. Later in 1958, this was successfully demonstrated at Holloman AFB by using a three-axis precision magnetometer attached to a plane and finding the best fit between the geomagnetic profile measured during the flight and the corresponding profile in a stored map. With these initiatives, a foundation for modern geomagnetic navigation was established.

Once terrain information was proven to be highly reliable for navigation, terrain maps constructed in previous years were used for conceptual proofs and testing. The company E-systems pioneered terrain navigation implementations and developed a successful terrain contour matching system known as TERCOM in 1973. The company conducted experiments using maps made in 1895, which were sufficiently similar to the vertical profiles measured at the test to achieve a match. Around the same concept, E-systems developed an abnormal contour matching system based on geomagnetism, called MAGCOM, and the use of geomagnetic anomaly data was carried out as a terrain map.

The main component covers more than 95% of the total magnitude while the component due to the anomalies rounds 4\% of the sum. The remaining 1% comes from disturbances, including diurnal variations. This magnitude became an interesting source of information because it has a unique characteristic. It can be described as a vector, with a direction and magnitude, and it can be decomposed along different axis (Goldenberg, 2006). Generally, the magnetic field vector is resolved under three components, along with the coordinate axis NED: Geographical North, East, and Downward direction perpendicular to the surface. The projection of the

magnetic field vector into the horizontal plane, called the horizontal component, always points to the geomagnetic north. The angle between the horizontal component and the geographical North is called magnetic declination. The angle between the magnetic vector and the horizontal plane is the magnetic inclination (Zhou 2021).

The key concept is to use the time-invariant characteristics of the crustal field to improve the estimation of the inertial navigation system (Canciani 2017) (Cuenca 2021). The use of the Earth magnetic field for position reference relies on the anomalies on the field due to the crustal rocks, generating a fingerprint of the area that can be represented by vertical contours of topography.

In geomagnetic matching, the previous existence of information about the test area's magnetic field is required to use correlation techniques for the matching process. Hence, a geomagnetic database is required before the execution of the algorithm. Currently, maps for the core and anomaly components are available for public use at the National Oceanic and Atmospheric Administration (NOAA) website, but these databases are suitable if high resolution is not a constraint for the navigation algorithm. For navigation in small local areas the geomagnetic map must be built experimentally from field measurements.

Cost – Rank 3

Geomagnetic navigation can be sensitive to the selection of the sensors. Geomagnetic Matching processes in areas with high features as local or indoor environments allow the use of low-cost sensor while it is recommended to use better quality sensor for outdoor applications. The prices my range from 15 USD as the LIS3MDL 3 Pololu compass or Pimoroni LSM303D, to 260 USD as the MGL SP-6 or 370 USD as the GMU 11 which is meant for industrial aircrafts.

Technical Readiness – Rank 1

The geomagnetic navigation is a promising alternative metho for navigations similar as terrain navigation currently under testing and research process. It has been proposed and tested by the Goodrich Corporation and currently is undergoing field testing performed by the US Airforce and academic research groups. Researchers have shown satisfactory results of geomagnetic navigation under favorable magnetic conditions as presented by the Airforce Institute of Technology.

Ease of Implementation/Use - Rank 2

Due to the novelty of the concepts and is development stage, it requires a mayor understanding for implementation of this type of algorithm since no open libraries and commercial devices provide yet this capability. Therefore, its implementations require trained personal capable of modifying the architecture of this type of system in all levels of complexity. This includes the geomagnetic database integrations, the guidance system adequacy to integrate the geomagnetic matching process and the corresponding algorithms and the measurement postprocessing to remove different sources of identified noises.

Size, Weight, and Power (SWaP) – Rank 4

The sensor weight and consumption powers is low. Models as the Garmin GMU 11 for industrial and heavy implementations weights 72 g with a power consumption of 1.4W for while the most common cases with UAVs developed with MEMS technologies have magnetometers as the LIS3MDL and similar with weights around 0.6 to 1 gram with power consumption of 10 to 100mW for a raking of 5.

Impact – Rank 4

The impact of this technology is considerable especially for cases where GPS technology is not available or weak since it provides with an additional sources of position information that do not relies on active signals. Its impact on position accuracy is low but its robustness under different weather, terrain and even and cybersecurity conditions suggests this methodology as a potential mitigation strategy for UAV navigation, especially in urban environments.

Assessment Score: 14-Medium

IV.3. Cellular Signal Navigation

Utilizing cellular networks, specifically 5G, has been a popular topic in relation to the future of UAS commercial operations. Where most current UAS regulations limit flight to visual line of sight (VLOS), and market-available drones primarily utilize radio frequencies as a command link, innovations and adoption of 5G cellular networks as a communication platform introduce new possibilities for beyond visual line of sight flight (BVLOS) and expanded operation in urban areas for the commercial UAS industry.

Furthermore, this provides existing and expanding infrastructure to enhance positioning of UAV navigation. A 5G networked drone can theoretically utilize the connected network base stations to produce a position solution that could supplement and improve the GNSS-based position, or act as a failsafe in areas of low GNSS availability or environment of higher RF interference.

The potential for 5G positioning techniques have been primarily explored in the context of similar applications utilizing 5G networks, like autonomous vehicle navigation. Some UAS-based positioning algorithms have been tested through simulation. Actual flight test case studies are sparse in the available literature and requires further exploration to realize the most accurate and safe version of commercial BVLOS UAS operation, especially in urban environments.

Cost – Rank 2

Experimental flight testing has included a software defined radio platform (SDR) such as an USRP E312 to facilitate the processing of cellular signals, and an antenna add-on, additionally. These components well exceed the $1000 threshold off the shelf (Abdallah & Kassas, 2021). However, 5G capable drones have been released and are projected to increase in availability as 5G infrastructure progresses (Qualcomm, 2021). Assuming antenna and capability are available off the shelf, additional cost to the UAS unit would be much lower.

Technical Readiness - Rank 3

Systems are still within the experimental phase, and BVLOS flight is not currently permitted. Case studies utilizing cell signal positioning demonstrate sub-meter accuracy potential in near urban environments based on flight testing (Abdallah & Kassas, 2021), (Shamaei & Kassas, 2018), (Khalife & Kassas, 2017). No commercial systems readily available, though cellular connectivity in market-available UAS is promising and projected. Infrastructure for UAS BVLOS 5G corridors are also being tested for operations in urban areas (sUAS News, 2021).

Ease of Implementation/Use – Rank 4

Open source options currently exist off the shelf for testing cellular positioning (4G) as analog for 5G-based navigation, and further adoption of 5G capabilities in market-available drones

will mean most hardware infrastructure available on UAS as is (Parrot, 2021) (Qualcomm, 2021). Further implementation for cellular-based positioning would be software related. Currently, off-the-shelf retrofitting of SDR and antenna onto UAS is most achievable (based on published methods of initial flight tests), with algorithm written or adapted for carrier phase filtering and processing of cell signals (4G/5G) (Abdallah & Kassas, 2021), (Shamaei & Kassas, 2018).

Size, Weight, and Power (SWaP) - Rank 2

Based off published testing, 446g SDR with internal battery of unknown weight, though very likely within the margins of the rank 2 listed <1kg weight (Abdallah & Kassas, 2021). Antenna component could vary in size while remaining under rank maximum weight. Similar components could weigh less but unlikely to reduce weight enough to meet Rank 3 specification levels. Further built-in availability of 5G capabilities in drones will likely improve this ranking, utilizing stock antenna and cellular signal processing capability.

Impact – Rank 4

Potential for UAS cell signal-based positioning as a supplement navigational system is promising, offering potentially submeter accuracy or better (Zeng, Y., 2020) (Abdallah & Kassas, 2021). Characteristics of urban areas include increased signal density and interference for UAS operations. Considering this fact, this method opportunistically utilizes one of those interference sources to diversify signal input and mitigate jamming/interference scenarios.

Effectiveness Score 15 – HIGH

IV.4. Wi-Fi Navigation

Utilizing Wi-Fi signal navigation as a failsafe or secondary system to inform a GNSS and IMU system onboard a UAS in the case of dropout or jamming in the urban environment could be viable, but limited by lack of published flight testing. Mitigation rating is based on foundation of literature documenting indoor UAS testing and non-UAS applications outdoors such as pedestrian navigation.

Cost – Rank 3

Signal antenna and processor needed to receive and localize the signals. A simplified approach in indoor navigation utilized a Raspberry Pi 3B+, ($40-$100). It is inferred a more robust (and costly) processor may be needed to work through a denser and more dynamic signal environment outdoors in the urban environment. Non-built in wifi antenna may be needed to expand signal receiving capability (Li, Sensors, and 2022, n.d.; Kapoor et al. n.d.; Raspberry 2021).

Technical Readiness - Rank 2

Literature on Wi-Fi positioning for pedestrian navigation in the urban canyon, and published tests of UAS Wi-Fi positioning in an indoor setting are available. Published testing of the system in the urban environment, on-board is sparse and would require further experimental trials. Conceptual phase (Cheng, et al., 2014) (Li and Zhang, 2022)

Ease of Implementation/Use – Rank 2

Modular addition of a processor with custom or selected software and an included Wi-Fi antenna, or separate antenna component, to be retrofitted on UAS. Hotspot database component to recognize and process Wi-Fi hotspots along the trajectory will need to be identified, loaded,

and regularly updated in preparation for operation. With no commercially available product, implementation and operation training necessary.

Size, Weight, and Power (SWaP) - Rank 3/4

Literature exploring indoor navigation used a Raspberry Pi 3B+, weighing about 60g with a built in Wi-Fi antenna. It's a possibility that outdoor navigation will require a more robust Wi-Fi antenna array. Assigned current rank with potential for similar or slightly heavier setup (Li and Zhang, 2022).

Impact – Rank 2

Utilizing Wi-Fi signal navigation as a failsafe or secondary system to inform a GNSS and IMU system onboard a UAS in the case of dropout or jamming in the urban environment could be viable based on indoor testing and non-UAS applications outdoors. Some potential challenges with this approach include the flying height of UAS operations and its relationship with vertically degrading signal strength of hotspots at ground level; The sourcing, updating, and storage needs of hotspot data for trilateration could cause an overestimate in the ratings listed above (Cheng, et al., 2014) (Kapoor, et al., 2014). Would be less effective as a mitigation technique in agricultural areas due to lower density of signal sources. Rank reduced by one due to lack of project-specific flight testing to draw conclusions.

Effectiveness Score 12 – MEDIUM

V. ECD, GPS and ADS-B Signal Spoofing Potential Mitigation Assessment

Many methods have been proposed to detect and mitigate GPS spoofing. The lion's share of the research focuses on detecting spoofing attacks. Methods of spoofing mitigation are often specialized or computational burdensome. This report highlights the brilliant value-added research by Dr. Manuel Eichelberger on the mitigation and recovery of GPS spoofing (Eichelberger 2019). ECD implementation and evaluation show that the robustness of collective detection (CD) can be exploited to mitigate spoofing attacks with some modifications. (Eichelberger 2019)shows that multiple locations, including the actual one, can be recovered from scenarios in which several signals are present. ECD does not track signals. It works with signal snapshots. It is suitable for snapshot receivers, which are a new class of low-power GPS receivers (Eichelberger 2019; H. Liu et al. 2007).

ADS-B's high dependency on communication and navigation (GNSS/ GPS) systems causes the system to inherit the vulnerabilities of those systems. This results in more opportunities (threats) to exploit those vulnerabilities. Advancements in computers, connectivity, storage, hardware, software, and apps are major aids to malicious parties who wish to carry out spoofing and other threats by exploiting the vulnerabilities of ADS-B. Another main vulnerability of ADS-B systems is their broadcast nature without security measures, which can easily be exploited to cause harm.

In this section, four primary concepts result from the investigation:

1) That UAS / drones are a mobile deployment agent. They are capable of Cyber-Spoofing navigation signals in the air by acting as a rogue access point, HAPs unit, mobile malicious signal generator, or interference medium to the ground control, friendly airborne unit, CBRN asset, or any other mechanism/system requiring localization or position fix via GPS / GNSS.

2) That GPS spoofing detection and mitigation for GNSS / GPS systems can be solved using the brilliant ECD algorithm for detection, mitigation, and recovery.

3) ADS-B is a subset of the larger receiver localization problem. Solutions that apply to the larger vector space, GNSS / GPS, also are valid for the subset, ADS-B, if computational hardware or cloud computing are available.

4) ECD Mitigation Assessment of ECD shows a cumulate score of 15 with extensive IMPACT and High Effectiveness. Further Stage 2 simulation work is cost-effective and highly recommended.

Definition: Spoofing - A Cyber-weapon attack that generates false signals to replace valid ones. GPS Spoofing is an attack to provide false information to GPS receivers by broadcasting counterfeit signals similar to the original GPS signal or by recording the original GPS signal captured somewhere else at some other time and then retransmitting the signal. The Spoofing attack causes GPS receivers to provide the wrong information about position and time (Humphreys et al., n.d.)(Tippenhauer et al. 2011). GPS Spoofed UAS / drones may deliver signals against any target (CBRN assets included) that requires accurate position fix or localization (Nichols, Sincavage, et al., n.d.).

It is important to understand that both GPS (part of the GNSS family) and ADS-B systems are vulnerable to spoofing attacks on both manned and unmanned aircraft. In general, GPS vulnerabilities translate down to the more specific ADS-B subset, which has vulnerabilities in its own right. This summary report details the brilliant work of Dr. Michael Eichelberger on *Robust Global Localization using GPS and Aircraft Signals*. He describes a functional tool known as CD to detect, mitigate and counter spoofing (and jamming) attacks on all stages of GPS.(Eichelberger 2019) The tool has been nicknamed ECD to honor Dr. Manuel Eichelberger's brilliant doctoral research. ECD is Dr. Manuel Eichelberger's advanced implementation of CD to detect and mitigate spoofing attacks on GPS or ADS-B signals

GPS is ubiquitous and is incorporated into so many applications (aircraft, ship, car /truck navigation; train routing and control; cellular network, stock market, CBRN assets, and power grid synchronization) that it makes a "rich" target for spoofing a receiver perceived location or time. Wrong information in time or space can have severe consequences.

ATC is partially transitioning from radar to a scheme in which aircraft (A/C) transmit their current location twice per second through ADS-B messages. This system has been mandated in Europe and has been well underway in the US since 2020. The A/C determine their location using GPS. If the onboard GPS receiver estimates a wrong location due to spoofing, wrong routing instructions will be delivered due to a wrong reported A/C location, leading to an A/C crash.

Ships depend heavily on GPS. They have few reference points to localize themselves apart from GPS. Wrong location indication can strand a ship, cause a collision, push off course into dangerous waters, ground a ship, or turn a ship into a ghost or a missile. In 2017 multiple incidents in the Black Sea and South China Seas have been documented (Nichols et al. 2019)(Nichols et al. 2019).

While planes and ships suffer spoofing attacks in the location domain, an attacker may also try to change the perceived time of a GPS receiver. Cellular networks rely on accurate time synchronization to exchange data packets between ground antennas and mobile handsets in the same network cell. Also, all neighboring network cells need to be time-synchronized for seamless call handoffs of handsets switching cells and coordinating data transmissions in overlapping coverage areas. Since most cellular ground stations get their timing information from GPS, a signal spoofing attacker could decouple cells from the common network time. Overlapping cells might send data simultaneously and frequencies, leading to message collisions and losses. Failing communications networks can disrupt emergency services and businesses. (Eichelberger 2019)

V.1. ECD Definitions

There are several definitions that are crucial in the discussion of the ECD method and are provided

SPOOFING - Threats and weaknesses show that large damages (even fatal or catastrophic) can be caused by transmitting forged GPS signals. False signal generators may cost only a few hundred dollars of software and hardware.

A GPS receiver computing its location wrongly or even failing to estimate any location can have different causes. Wrong localization solutions come from 1) a low signal-to-noise ratio

(SNR) of the signal (examples: inside a building or below trees in a canyon); 2) reflected signals in multipath scenarios, or 3) deliberately spoofed signals.(Eichelberger 2019) discusses mitigating low SNR and multipath reflected signals. Signal spoofing (#3) is the most difficult case since the attacker can freely choose each satellite's signal power and delay individually.

GPS SYSTEM & SIGNALS - The GPS system and signals are well documented. See("2020-SPS-Performance-Standard," n.d.)

CLASSIC RECEIVERS - Classical GPS receivers use three stages when obtaining a location fix. They are Acquisition, Tracking, and Localization.

Acquisition. The relative speed between satellite and receiver introduces a significant Doppler shift to the carrier frequency. GPS receiver locates the set of available satellites. This is achieved by correlating the received signal with satellites' known C/A codes. Since satellites move at considerable speeds. The signal frequency is affected by a Doppler shift. So, the receiver must correlate the received signal with C/ A codes with different Doppler shifts.

Tracking. After a set of satellites has been acquired, the data contained in the broadcast signal is decoded. Doppler shifts and C /A code phase are tracked using tracking loops. After the receiver obtains the ephemeris data and HOW timestamps from at least four satellites, it can compute its location.

Localization. Localization in GPS is achieved using signal time of flight (ToF) measurements. TVs are the difference between the arrival times of the HOW timestamps decoded in the tracking stage of the receiver and those signal transmission timestamps themselves. The local time at the receiver is unknown, and the localization is done using pseudo ranges. The receiver location is usually found using least-squares optimization.

A main disadvantage of GPS is the low bit rate of the navigation data encoded in the signals transmitted by the satellites. The minimal data necessary to compute a location fix, which includes the ephemerides of the satellites, repeats only every 30 seconds (Eichelberger 2019).

A-GPS (ASSISTED GPS) – REDUCING THE STARTUP TIME - Assisted GPS (A-GPS) drastically reduces the startup time by fetching the navigation data over the Internet, commonly connecting via a cellular network. Data transmission over cellular networks is faster than decoding the GPS signals and normally only takes a few seconds. The ephemeris data is valid for 30 minutes. That data can reduce the acquisition time since the available satellites and their expected Doppler shifts can be estimated. The receiver still needs to extract the HOW timestamps from the signal with A-GPS. However, these timestamps are transmitted every six seconds, which translates to how long the A-GPS receiver takes to compute a location fix (Eichelberger 2019).

COURSE – TIME NAVIGATION - Course-Time Navigation (CTN) is an A-GPS technique that drops the requirement to decode the HOW timestamps from the GPS signals. The only information used from the GPS signals is the phases of the C/A code sequences detected by a matched filter. Those C/A code arrival times are unambiguously related to the sub-milliseconds; the deviation may be no more than 150 km from the correct values. Since the PRN sequences repeat every millisecond, without considering navigation data flips in the signal, CTN can, in theory, compute a location from one millisecond of the sampled signal. Noise can be an issue with such short signal recordings because it cannot be filtered

out the same way with longer recordings of several seconds. The big advantage is that signal processing is fast and power-efficient and reduces the latency of the first fix. Since no metadata is extracted from the GPS signal, CTN can often compute a location even in the presence of noise or attenuation (van Diggelen 2009).

SNAPSHOT RECEIVERS - Snapshot receivers aim at the remaining latency that results from the transmission of timestamps from satellites every six seconds. Snapshot receivers can determine the satellite modulo 1 ms ranges, which corresponds to 300 km.(Eichelberger 2019)

COLLECTIVE DETECTION - Collective Detection (CD) is a *maximum likelihood snapshot receiver localization method*, which does not determine the arrival time for each satellite but rather combines all the available information and decides only at the end of the computation. This technique is critical to the (Eichelberger 2019) invention to mitigate spoofing attacks on GPS or ADS-B. CD can tolerate a few low-quality satellite signals and is more robust than CTN. CD requires a lot of computational power. CD can be sped up by a branch and bound approach, which reduces the computational power per location fix to the order of one second even for uncertainties of 100 km and a minute.

ECD - Returning to Dr. Manuel Eichelberger's CD – Collective detection maximum likelihood localization approach, his method can detect spoofing attacks and mitigate them. [i]The ECD approach is a robust algorithm to mitigate spoofing. ECD can differentiate closer differences between the correct and spoofed locations than previously known approaches. COTS has little spoofing integrated defenses. Military receivers use symmetrically encrypted GPS signals, subject to a "replay" attack with a small delay to confuse receivers.

ECD solves even the toughest type of GPS spoofing attack, consisting of spoofed signals with power levels similar to the authentic signals. ECD achieves median errors under 19 m on the TEXBAT dataset, the de-facto reference dataset for testing GPS anti-spoofing algorithms (Ranganathan, Ólafsdóttir, and Capkun 2016). FOR EACH LOCATION FIX, the ECD approach uses only a few milliseconds' worth of raw GPS signals, so-called snapshots. This enables offloading the computation into the Cloud, allowing knowledge of observed attacks. Existing spoofing mitigation methods require a constant stream of GPS signals and tracking those signals over time. Computational load increases because fake signals must be detected, removed, or bypassed.

Researchers have been trying to find a complete solution to the spoofing threat because of the overwhelming dependence on GPS in every sector, ranging from civilian to military. Haider and Khalid 2016 published an adequate survey of spoofing countermeasures up through 2016 (Haider and Khalid 2016).

V.2. GPS Spoofing Techniques

There are three common GPS spoofing techniques with different sophistication levels. They are simplistic, intermediate, and sophisticated.

The *simplistic spoofing attack* is the most commonly used technique to spoof GPS receivers. It only requires a COTS GPS signal simulator, amplifier, and antenna to broadcast signals to the GPS receiver. It was performed successfully by Los Almos National Laboratory in 2002 (Warner and Johnston 2003). Simplistic spoofing attacks can be expensive as the GPS

simulator can run $400K and is heavy (not mobile). The available GPS signal does not synchronize simulator signals, and detection is easy.

In the *intermediate spoofing attack*, the spoofing component consists of a GPS receiver to receive a genuine GPS signal and a spoofing device to transmit a fake GPS signal. The idea is to estimate the target receiver antenna position and velocity and then broadcast a fake signal relative to the genuine GPS signal. This spoofing attack is difficult to detect and can be partially prevented by using an IMU (Humphreys et al., n.d.).

In *sophisticated spoofing attacks*, multiple receiver-Spoofer devices target the GPS receiver from different angles and directions. The angle-of-attack defense against GPS spoofing in which the angle of reception is monitored to detect spoofing fails in this scenario. The only known defense successful against such an attack is cryptographic authentication (Humphreys et al., n.d.).

Note that prior research on spoofing was to *exclude* the fake signals and focus on a single satellite. ECD *includes* the fake signal on a minimum of four satellites and then progressively / selectively eliminates their effect until the real *weaker* GPS signals become apparent (Eichelberger 2019).

Haider's detailed research on the above three attacks is available in (Haider and Khalid 2016)

V.3. GPS SPOOFING RESEARCH

Three research tracks are most relevant to ECD / CD**:** Maximum Likelihood Localization, Spoofing Mitigation algorithms, and Successive Signal Interference Cancellation (SIC). Note that historical spoofing research focuses primarily on detecting singular SPS source attacks. The focus on mitigation, correction, and recovery attending to multiple spoofing signals on multiple satellite attack surfaces is the hallmark of ECD.

CD is a maximum likelihood GPS localization technique. It was proposed in 1996 but considered computationally infeasible at that time. The search space contained millions or more location hypotheses. Improvements in the computational burden were found using various heuristics. A breakthrough came with the proposal of a branch-and-bound algorithm that finds the optimal solution within ten seconds running on a single CPU thread (Bissig, Eichelberger, and Wattenhofer, n.d.).

Most GPS spoofing defenses focus on detecting spoofing attacks. There is a lack of prior research for spoofing mitigation and recovering from successful attacks by finding and authenticating the correct signals. In contrast to the extensive research on GPS spoofing, there is a lack of commercial, civil receivers with anti-spoofing capabilities. ECD inherently mitigates spoofing attacks.

Spoofing hardware performing a *sophisticated, seamless satellite-lock takeover* attack has been built. Challenges associated with spoofing are matching the spoofed and accurate signals ' amplitudes at the receiver, which might not be in LOS and moving (Humphreys et al., n.d.).

It is practically feasible for a Spoofer to erase the authentic signals at a 180-degree phase offset. This is one of the strongest attacks that can only be detected with multiple receiver antennas or a moving receiver. The Spoofer needs to know the receiver location more accurately than the

GPS L1 wavelength, 19 cm, to make signal erasure feasible. Receivers with only a single antenna cannot withstand such an erasure attack. ECD targets single-antenna receivers and does not deal with signal erasure. The original signals are still present in all other spoofing attacks, including signal replay and multiple transmission antenna implementations, and ECD remains robust. Detecting multi-antenna receivers and differentiating signal timing consistencies are covered in (Tippenhauer et al. 2011).

The GPS anti-spoofing work most relevant to ECD is based on the joint processing of satellite signals and the maximum likelihood of localization.

ECD uses an iterative signal damping technique with spoofing signals similar to SIC. SIC removes the strongest received signals to find the weaker signals and has been used with GPS signals before. Previous work is based on a classical receiver architecture which only keeps a signal's timing, amplitude, and phase. The ECD has its snapshot receiver based on CD, which directly operates in the localization domain and does not identify individual signals in an intermediate stage. It is impossible to differentiate between authentic and spoofed signals, *a priori*, ECD does not remove signals from the sample data. Otherwise, the localization algorithm might lose the information from authentic signals/ Instead, ECD dampens strong signals by 60% to reveal weaker signals. This can reveal localization solutions with lower CD likelihood (Eichelberger 2019).

V.4. GPS Signal Jamming

The easiest way to prevent a receiver from finding a GPS location is by jamming the GPS frequency band. GPS signals are weak and require sophisticated processing to be found. Satellite signal jamming considerably worsens the satellite signal acquisition results' signal-to-noise ratio (SNR). ECD algorithms achieve a better SNR than classical receivers and tolerate more noise or stronger jamming (Eichelberger 2019).

A jammed receiver is less likely to detect spoofing since the original signals cannot be accurately determined. The receiver tries to acquire any satellite signals it can find. The attacker only needs to send a set of valid GPS satellite signals stronger than the noise floor without synchronizing with authentic signals (Eichelberger 2019).

There is a more powerful and subtle attack on top of the jammed signal. The Spoofer can send a set of satellite signals with adjusted power levels and synchronize to the authentic signals to successfully spoof the receiver. So even if the receiver has countermeasures to differentiate the jamming, the Spoofer signals will be accepted as authentic (Nichols, Mumm Wayne D Lonstein Julie JCH Ryan Candice M Carter, and Jch, n.d.; Nichols, Mumm, et al., n.d.).

Two of the most powerful GPS signal spoofing attacks are *Seamless Satellite-Lock Takeover (SSLT) and Navigation Data Modification (NDM).* ECD performance is assessed in each case.

The most powerful attack is a *seamless satellite-lock takeover*. The original and counterfeit signals are nearly identical in such an attack concerning the satellite code, navigation data, code phase, transmission frequency, and received power. This requires the attacker to know the location of the spoofed device precisely so that ToF and power losses over a distance can be factored in. After matching the spoofed signals with the authentic ones, the Spoofer can send its signals with a small power advantage to trick the receiver into tracking those instead of the authentic signals. A classical receiver without spoofing countermeasures, like tracking multiple

peaks, cannot mitigate or detect the SSLT attack, and there is no indication of interruption of the receiver's signal tracking (Eichelberger 2019).

The Navigation Data Modification (NDM) method is when an attacker has two attack vectors: modifying the signal's code phase or *altering the navigation data*—the former changes the signal arrival time measurements. The latter affects the perceived satellite locations. Both influence the calculated receiver location. ECD works with snapshot GPS receivers and is not vulnerable to NDM changes as they fetch information from other sources like the Internet. ECD deals with modified, wireless GPS signals (Eichelberger 2019).

V.5. ECD ALGORITHM DESIGN

ECD is aimed at single-antenna receivers. Its spoofing mitigation algorithm object is to identify all likely localization solutions. It is based on CD because 1) CD has improved noise tolerance compared to classical receivers, 2) CD is suitable for snapshot receivers, 3) CD is not susceptible to navigation data modifications, and 4) CD computes a location likelihood distribution which can reveal all likely receiver locations including the actual location, independent of the number of spoofed and multipath signals. ECD avoids spoofing pitfalls and signal selection problems by joining and transforming all signals into a location likelihood distribution. Therefore, it defeats the top two GPS spoofing signal attacks (Eichelberger 2019).

Relating to the 4th point, spoofing and multipath signals are similar from a receiver's perspective. Both result in several observed signals from the same satellite. The difference is that multipath signals have a delay-dependent on the environment while spoofing signals can be crafted to yield consistent localization solutions at the receiver. Classical receivers can be modified to track an arbitrary number of signals per satellite instead of only one to detect spoofing and multipath signals. The set of authentic signals – one signal from each satellite – would have to be correctly identified in such a receiver. Any selection of signals can be checked for consistency by verifying that the resulting residual error of the localization algorithm is very small. This is a combinatorically difficult problem. For n satellites and m transmitted sets of spoofed signals, there are $(m+1)^n$ possibilities for the receiver to select a set of signals. Only m + 1 of those will result in a consistent localization solution representing the actual location and m spoofed locations. ECD avoids this signal selection problem by joining and transforming all signals into a location likelihood distribution (Eichelberger 2019).

ECD only shows consistent signals since just a few signals overlapping (synced) for some location hypotheses do not significantly accumulate. All plausible receiver locations – given the observed signals - have a high likelihood. Finding these locations in four dimensions, space and time, is computationally expensive (Bissig, Eichelberger, and Wattenhofer, n.d.).

A fast CD leveraging branch and bound algorithm is employed to reduce the computational load compared to exhaustively enumerating all the location hypotheses in the search space. (Eichelberger 2019)describes the modifications to the B&B algorithm for ECD in copious detail in chapter 6.

One of the key points under the receiver implementation concerns the correlation of C/A codes. The highest correlation is theoretically achieved when the C/A code in the received signal is aligned with the reference C/A code. Due to the pseudo-random nature of the C/A codes, a shift larger than one code chip from the correct location results in a low correlation value. Since one code chip has a duration of 1/1023 ms, the width of the peaks found in the acquisition

vector is less than 2% of the total vector size. ECD reduces the maximum peak by 60% in each vector. A detection for partially overlapping peaks prevents changes to those peaks. Reducing the signal rather than eliminating it has a little negative impact on the accuracy. Before using these vectors in the next iteration of the algorithm, the acquisition result vectors are normalized again. This reduces the search space based on the prior iteration.

The ECD mitigation workflow for detection and mitigation of spoofing of GNSS with Collective Detection approach from Dr. Manuel Eichelberger is as follows:

Step 1: receiver localization in space (3D space and time)

Step 2: assessing pseudo-likelihood of the receiver location in the Collective Detection approach

Step 3: Detection of spoofing, GNSS information is compromised, e.g., multiple locations present

Step 4: determine which of the likely locations is correct, i.e., mitigate the spoofing.

V.6. SIGNAL SPOOFING

ADS-B signal spoofing attempts to deceive an ADS-B receiver by broadcasting fake ADS-B signals that resemble a set of normal ADS-B signals or by re-broadcasting genuine signals captured elsewhere or at a different time. Spoofing an ADS-B system is also known as message injection because fake (ghost) a/c is introduced into the air traffic. The system's vulnerability – having no authentication measures implemented at the systems data link layer – enables this threat. Spoofing is a hit on the security goal of Integrity. This leads to undesired operational decisions by controllers or surveillance operations in the air or on the ground. The threat affects both ADS-B IN and OUT systems. Spoofing threats are two basic varieties: Ground Station Target Ghost Injection / Flooding and Ground Station Target Ghost Injection / Flooding (ALI 2019).

Ground Station Target Ghost Injection / Flooding is performed by injecting ADS-B signals from a single a/c or multiple fakes (ghost) a/c into a ground station. This will cause single /multiple fake (ghost) a/c to appear on the controller’s working position (radar screen) (ALI 2019).

Aircraft Target Ghost Injection / Flooding is performed by injecting ADS-B signals from a single a/c or multiple fake (ghost) a/c into an airplane in flight. This will cause ghost a/c to appear on the TCAS and CDTI screens in the cockpit to go haywire. Making the mess worse, the fake data will also be used by airborne operations such as ACAS, ATSAW, ITP, and others for aiding a/c navigation operations (ALI 2019).

An a/c can look like it has vanished from the ADS-B-based air traffic by deleting the ADS-B message broadcast from the a/c. This can be done by two methods: destructive interference and constructive interference. Destructive interference is performed by transmitting an inverse of an actual ADS-B signal to an ADS-B receiver. Constructive interference is performed by transmitting a duplicate of the ADS-B signal and adding the two signal waves (original and duplicate). The two signal waves must be of the same frequency phase and traveling in the

same direction. Both approaches will result in being discarded by the ADS-B receiver as corrupt (ALI 2019).

ADS-B message modification is feasible on the physical layer during transmission via datalinks using two methods: Signal Overshadowing and Bit-flipping. Signal overshadowing is done by sending a stronger signal to the ADS-B receiver, whereby only the stronger of the two colliding signals is received. This method will replace either the whole target message or part of it. Bit flipping is an algorithmic manipulation of bits. The attacker changes bits from 1 to 0 or vice versa. This will modify the ADS-B message and is a clear violation of the security goal of Integrity. This attack will disrupt ATC operations or a/c navigation.

V.7. ECD Performance Assessment

The proposed assessment metrics assess the overall effectiveness of mitigation schemes. Five parameters are evaluated to quantify the overall score to rank the proposed methods. These factors are:

1.) Cost
2.) Technical Readiness
3.) Ease of Implementation/Use
4.) Size, Weight, and Power (SWaP)
5.) Impact

Each factor will be ranked with a numerical score from 1 to 5, with 1 being the "worst" and 5 being the "best" in each category. A detailed guide for each ranking factor is provided based on implementing the mitigation scheme on a small UAS. The ranking for each factor is provided for the ECD method.

Cost Ranking - 3
Technical Readiness - 2
Ease of Implementation/Use - 3
Size, Weight, and Power (SWaP) - 3
Impact - 5

Effectiveness score: 16 High

The purposes of this section were to introduce the problem of Navigation Cyber-Spoofing; to recognize that GNSS / GPS / ADS-B systems, including CBRN mobile assets, are susceptible to Cyber Spoofing; that research has focused on detection rather than mitigation and recovery efforts; that ECD is a brilliant solution to part of the Cyber Spoofing problem as it does not exclude false signals but encompasses them into the algorithm; and lastly that in terms of the ASSURE44 mitigation schema, ECD Mitigation Assessment of ECD shows a cumulate score of 15 with extensive IMPACT and High Effectiveness. Further Stage 2 simulation work is cost-effective and highly recommended.

VI. Summary and Conclusions

This Identification of Potential Mitigations report fulfills Task 2 for the A44 ASSURE project. Examination of recorded ABS-B data was conducted to expose potential risks and provide guidance on mitigation schemes. The examination reveals dropouts and anomalies that occur in flight operations. Based on the risk assessments in Task 1, the performer conducted a market survey of market solutions to mitigate loss of GPS and loss of ADS-B data as well as a market survey of market solutions to mitigate unvalidated GPS and unvalidated ADS-B In data. The market surveys include estimated costs, ease of implementation, and a preliminary assessment of the effectiveness of market solutions to mitigate the various risks identified in Task 1.

The integrity of navigation systems, such as ADS-B and GPS, was analyzed to detect threats to the integrity. These risks include erroneous, spoofed, jammed, and dropped data from GPS or ADS-B systems. Recorded ABS-B data was examined to expose potential risks and provide guidance on mitigation schemes. It reveals dropouts and anomalies that occur during flight operations. Two primary data set types were used in this study: GPS data from the Dallas Fort Worth Airport and data from the OpenSky Network. The results are informative and provide real-world assessment of GPS and ADS-B navigation data. Thereby providing knowledge of how often and for how long dropouts and other erroneous data are occurring. The type of machine learning algorithms and the associated settings that process the data more efficiently and effectively was studied. In the studies it was observed that for sUAV flights the average maximum altitude was 375 ft., a typical time delay after outlier removal was approximately 3.5 seconds, and the average upper bound for determining dropout instances was near 6.25 seconds.

Several mitigation schemes were evaluated for their effectiveness in jamming and spoofing conditions. The mitigation schemes evaluated were optical flow, geomagnetic navigation, cellular signal navigation, WIFI navigation, and Eichelberger's Collective Detection (ECD) method and the findings are summarized in Table 10.

Table 10. Summary of the GPS and ADS-B risk mitigation methods

Mitigations Scheme	Condition	Assessment Score	Effectiveness
AI Path Prediction	Drop Outs	13	Medium
Optical Flow	Jamming	16	High
Geomagnetic Navigation	Jamming	14	Medium
Cellular Signal Navigation	Jamming	15	High
W-Fi Navigation	Jamming	12	Medium
ECD	Spoofing	16	High

The study of these five systems indicate that most have an overall high effectiveness rating, while having varying effectiveness in each of the five factors scored. It should be noted that additional mitigation strategies were briefly reviewed but were not of sufficient interest by the team to include in the full evaluation.

It is the A44 team opinion that flight and simulation testing should continue on all 5 of the mitigation methods and continued efforts be made in identifying dropouts and erroneous data in the current data sets along with new data sets obtained.

VII. REFERENCES

"2020-SPS-Performance-Standard." n.d.

Abdallah, A A, Z M Kassas - Proceedings of the 34th International Technical, and undefined 2021. 2021. "UAV Navigation with 5G Carrier Phase Measurements." *Ion.Org*. https://www.ion.org/publications/abstract.cfm?articleID=18101.

Abdallah, Ali A., and Zaher M. Kassas. 2021a. "UAV Navigation with 5G Carrier Phase Measurements." *Ion.Org*. https://www.ion.org/publications/abstract.cfm?articleID=18101.

2021b. "UAV Navigation with 5G Carrier Phase Measurements." *Ion.Org*. https://www.ion.org/publications/abstract.cfm?articleID=18101.

Airport Authority of India. 2022. "What Are NOTAMS?" 2022. https://www.aai.aero/en/content/what-notam.

ALI, BUSYAIRAH SYD. 2019. *AIRCRAFT SURVEILLANCE SYSTEMS : Radar Limitations and the Advent of the Automatic Dependent... Surveillance Broadcast.* ROUTLEDGE.

Bissig, Pascal, Manuel Eichelberger, and Roger Wattenhofer. n.d. "Fast and Robust GPS Fix Using One Millisecond of Data." www.disco.ethz.ch.

Bottou, Léon, and Chih-Jen Lin. n.d. "Support Vector Machine Solvers."

Buitinck, Lars, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, et al. 2013. "API Design for Machine Learning Software: Experiences from the Scikit-Learn Project," September. http://arxiv.org/abs/1309.0238.

Cheng, J, L Yang, Y Li, W Zhang - Physical Communication, and undefined 2014. n.d. "Seamless Outdoor/Indoor Navigation with WIFI/GPS Aided Low Cost Inertial Navigation System." *Elsevier*. https://www.sciencedirect.com/science/article/pii/S1874490714000020?casa_token=wBI7ob47LBcAAAAA:RPF-l-FmR7YKm8eSCq3bdSj7aVTfMpcFva91Ed9irahznLvhGktpH14G_ORpSU4Nrg_FHm2q9aM.

Darabseh, Ala ', Evangelos Bitsikas, and Brice Tedongmo. 2019. "EPiC Series in Computing Detecting GPS Jamming Incidents in OpenSky Data." Vol. 67.

DECS Research. 2022. "OpenSky Github Repository."

Diggelen, Frank Stephen Tromp. van. 2009. *A-GPS : Assisted GPS, GNSS, and SBAS*. Artech House.

Eichelberger, Manuel. 2019. "ETH Library Robust Global Localization Using GPS and Aircraft Signals." https://doi.org/10.3929/ethz-b-000379990.

estaff. 2021. "Address Military GPS Jamming." Aviation Safety. 2021.

"Ettus Research." n.d. Https://Www.Ettus.Com/All-Products/USRP-E312/.

Federal Aviation Administration. 2016. "Public ADS-B Performance Report (PAPR) User ' s Guide Flight Standards Service Background – Public ADS-B Performance Report," no. March: 1–18.

Haider, Zeeshan, and Shehzad Khalid. 2016. “Survey on Effective GPS Spoofing Countermeasures.” *Journal of E-Technology*. Vol. 7.

Harris, Mark. 2021. “Military Tests That Jam and Spoof GPS Signals Are an Accident Waiting to Happen.” *IEEE Spectrum* 58 (2): 22–27. https://doi.org/10.1109/MSPEC.2021.9340116.

Humphreys, Todd E, Mark L Psiaki, Brady W O’hanlon, and Paul M Kintner. n.d. “Assessing the Spoofing Threat: Development of a Portable GPS Civilian Spoofer.” http://philosecurity.org/2008/09/07/gps-spoofing.

IEEE Spectrum. n.d. “FAA FILES REVEAL A SURPRISING THREAT TO AIRLINE SAFETY: THE U.S. MILITARY’S GPS TESTS.”

Kapoor, R, S Ramasamy, A Gardi, R Sabatini - Energy Procedia, and undefined 2017. n.d. “UAV Navigation Using Signals of Opportunity in Urban Environments: A Review.” *Elsevier*. Accessed June 18, 2022. https://www.sciencedirect.com/science/article/pii/S1876610217301868.

Khalife, J J, S Bhattacharya, … Z M Kassas - Meeting of the Satellite Division of, and undefined 2018. n.d. “Centimeter-Accurate UAV Navigation with Cellular Signals.” *Ion.Org*. https://www.ion.org/publications/abstract.cfm?articleID=16105.

Kumar, Krishan, and Ranjan Kumar Gupta. 2018. “Signal to Noise Ratio Based Wi-Fi Offloading Decision Algorithm in Vehicular Networks.” In *Procedia Computer Science*, 125:910–16. Elsevier B.V. https://doi.org/10.1016/j.procs.2017.12.116.

Li, Z, Y Zhang - Sensors, and undefined 2022. n.d. “Constrained ESKF for UAV Positioning in Indoor Corridor Environment Based on IMU and WiFi.” *Mdpi.Com*. https://www.mdpi.com/1436582.

Liu, Hui, Houshang Darabi, Pat Banerjee, and Jing Liu. 2007. “Survey of Wireless Indoor Positioning Techniques and Systems.” *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*. https://doi.org/10.1109/TSMCC.2007.905750.

Liu, Z, S Lo, T Walter - Proceedings of the 33rd International Technical, and undefined 2020. n.d. “Characterization of ADS-B Performance under GNSS Interference.” *Ion.Org*. Accessed June 18, 2022. https://www.ion.org/publications/abstract.cfm?articleID=17675.

Murrian, Matthew J., Lakshay Narula, Peter A. Iannucci, Scott Budzien, Brady W. O’Hanlon, Mark L. Psiaki, and Todd E. Humphreys. 2020. “First Results from Three Years of GNSS Interference Monitoring from Low Earth Orbit,” September. http://arxiv.org/abs/2009.04093.

“New York’s 50-Mile Drone Corridor Integrates 5G Test Network.” n.d. Https://Www.Suasnews.Com/2021/09/New-Yorks-50-Mile-Drone-Corridor-Integrates-5g-Test-Network/.

Nichols, Randall K, Hans C Mumm, Wayne D Lonstein, Julie JCH Ryan, Candice Carter, and Julie Jch. 2019. “Unmanned Aircraft Systems in the Cyber Domain.” https://newprairiepress.org/ebooks.

n.d. “Counter Unmanned Aircraft Systems Technologies and Counter Unmanned Aircraft Systems Technologies and Operations Operations.” https://newprairiepress.org/ebooks.

Nichols, Randall K, Hans C Mumm Wayne D Lonstein Julie JCH Ryan Candice M Carter, and Julie Jch. n.d. “Unmanned Vehicle Systems & Operations on Air, Sea, Land.” https://newprairiepress.org/ebooks.

Nichols, Randall K, Suzanne Sincavage, Hans Mumm, Wayne Lonstein, Candice Carter, Randall K ; Nichols, Suzanne ; Sincavage, et al. n.d. “DRONE DELIVERY OF CBNRECy-DEW WEAPONS Emerging DRONE DELIVERY OF CBNRECy-DEW WEAPONS Emerging Threats of Mini-Weapons of Mass Destruction and Disruption Threats of Mini-Weapons of Mass Destruction and Disruption (WMDD) (WMDD).” https://newprairiepress.org/ebooks.

Parrot. 2021. “No Title.” Parrot ANAFI Ai 4G Technical Documentation. (2021). 2021.

Pedregosa FABIANPEDREGOSA, Fabian, Vincent Michel, Olivier Grisel OLIVIERGRISEL, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Jake Vanderplas, et al. 2011. “Scikit-Learn: Machine Learning in Python Gaël Varoquaux Bertrand Thirion Vincent Dubourg Alexandre Passos PEDREGOSA, VAROQUAUX, GRAMFORT ET AL. Matthieu Perrot.” *Journal of Machine Learning Research*. Vol. 12. http://scikit-learn.sourceforge.net.

Qualcomm. 2022. “No Title.” *Qualcomm Unleashes a New Era of Autonomous Drone Capabilities with World’s First 5G and AI-Enabled Drone Platform*, https://www.qualcomm.com/news/releases/2021/08/qua.

Ranganathan, Aanjhan, Hildur Ólafsdóttir, and Srdjan Capkun. 2016. “SPREE: A Spoofing Resistant GPS Receiver.” In *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, 0:348–60. New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/2973750.2973753.

Raspberry. 2021. “No Title,” https://www.raspberrypi.com/products/raspberry-pi-.

Schäfer, Matthias, Martin Strohmeier, Vincent Lenders, Ivan Martinovic, and Matthias Wilhelm. 2014. “Bringing up OpenSky: A Large-Scale ADS-B Sensor Network for Research.” In *IPSN 2014 - Proceedings of the 13th International Symposium on Information Processing in Sensor Networks (Part of CPS Week)*, 83–94. IEEE Computer Society. https://doi.org/10.1109/IPSN.2014.6846743.

Semke, William, Nicholas Allen, Asma Tabassum, Matthew McCrink, Mohammad Moallemi, Kyle Snyder, Evan Arnold, Dawson Stott, and Michael G Wing. n.d. “Analysis of Radar and ADS-B Influences on Aircraft Detect and Avoid (DAA) Systems.” *Mdpi.Com*. Accessed June 18, 2022. https://doi.org/10.3390/aerospace4030049.

Shamaei, K, Z M Kassas - Proceedings of the 32nd International Technical, and undefined 2019. n.d. “Sub-Meter Accurate UAV Navigation and Cycle Slip Detection with LTE Carrier Phase Measurements.” *Ion.Org*. https://www.ion.org/publications/abstract.cfm?articleID=17051.

Shamaei, Kimia, and Zaher M. Kassas. n.d. “Sub-Meter Accurate UAV Navigation and Cycle Slip Detection with LTE Carrier Phase Measurements.” *Ion.Org*. Accessed June 18, 2022. https://www.ion.org/publications/abstract.cfm?articleID=17051.

Shaukat, SA, K Munawar, M Arif, … AI Bhatti - … on Intelligent and, and undefined 2016. n.d. “Robust Vehicle Localization with GPS Dropouts.” *Ieeexplore.Ieee.Org*. Accessed June 18, 2022. https://ieeexplore.ieee.org/abstract/document/7824135/?casa_token=djhUQWz2sLEAA

AAA:1UMcMFX4zz0OvIDvI0psy6Cc6uhKL-nBnYkb5_wbBErRvIHrE4Z5iidym0xTvK4FYVp0PadghA.

Simon, Hans Ulrich, and Nikolas List. 2009. “SVM-Optimization and Steepest-Descent Line Search. SVM-Optimization and Steepest-Descent Line Search *.” https://www.researchgate.net/publication/221497771.

Sun, Junzi. n.d. “The 1090 Megahertz Riddle A Guide to Decoding Mode S and ADS-B Signals.”

Sun, K, Y Yu, W Zhou, … G Zhou - 2013 IEEE International, and undefined 2013. n.d. “A Low-Cost and Robust Optical Flow CMOS Camera for Velocity Estimation.” *Ieeexplore.Ieee.Org*. Accessed June 18, 2022. https://ieeexplore.ieee.org/abstract/document/6739624/.

Tabassum, Asma, and William Semke. 2018a. “Assessing the Effect of ADS-B Message Drop-out in Detect and Avoid of Unmanned Aircraft System Using Monte Carlo Simulation.” *Safety* 4 (4). https://doi.org/10.3390/safety4040049.

———. 2018b. “Assessing the Effect of ADS-B Message Drop-Out in Detect and Avoid of Unmanned Aircraft System Using Monte Carlo Simulation.” *Safety* 4 (4): 49. https://doi.org/10.3390/safety4040049.

Tippenhauer, Nils Ole, Christina Pöpper, Kasper B. Rasmussen, and Srdjan Čapkun. 2011. “On the Requirements for Successful GPS Spoofing Attacks.” In *Proceedings of the ACM Conference on Computer and Communications Security*, 75–85. New York, New York, USA: ACM Press. https://doi.org/10.1145/2046707.2046719.

U.S. Department of Transporation. 2022a. “Positioning, Navigation and Timing (PNT) Program.” 2022.

Warner, J. S., and R. Johnston. 2003. “GPS Spoofing Countermeasures.”