Automatic Dependent Surveillance-Broadcast Preamble Classification for Spoofing Detection

Darren Kah Hou Quek[1], Guang Hua[2*], Zhiping Lin[1]
[1]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
E-mail: kquek009@e.ntu.edu.sg, ezplin@ntu.edu.sg
[2]Infocomm Technology Cluster, Singapore Institute of Technology, Singapore
E-mail: guang.hua@singaporetech.edu.sg

***Abstract*—This paper addresses the security vulnerabilities in Automatic Dependent Surveillance–Broadcast (ADS-B), a pivotal technology in modern air traffic management. While ADS-B enhances situational awareness and operational efficiency, its reliance on open, unencrypted transmissions renders it susceptible to spoofing, jamming, and eavesdropping. Focusing on anti-spoofing techniques, this paper introduces a novel approach to classifying spoofed versus authentic preamble of ADS-B signals using machine learning methods, including k-Nearest Neighbors (kNN) and Multilayer Perceptron (MLP). We curated a dataset of 255 signal traces under high-noise conditions for the experiment, exploring the effectiveness of steady-state amplitude and transient-state correlation analyses alongside the proposed classifiers. The results reveal that while MLP models achieve promising precision rates of over 90%, optimization and larger datasets are required to meet stringent safety standards. These findings offer a foundation for developing robust, real-time ADS-B spoofing detection systems, critical for ensuring aviation safety.**

I. Introduction

Automatic Dependent Surveillance–Broadcast (ADS-B) is a cornerstone technology in contemporary air traffic management, revolutionizing situational awareness and enhancing flight safety. By leveraging satellite navigation, ADS-B enables aircraft to autonomously determine their position and broadcast it periodically. This real-time transmission of location, velocity, and other flight parameters provides unprecedented visibility for ground stations and other aircraft. As a foundational element of both the Next Generation Air Transportation System (NextGen) and the Single European Sky ATM Research (SESAR) program [1], ADS-B plays a pivotal role in transforming airspace management by enhancing efficiency and reducing reliance on traditional radar-based surveillance.

The widespread adoption of ADS-B delivers substantial advantages. This includes increased airspace safety, improved monitoring, and operational efficiency [2]. However, the system's dependence on open, unencrypted transmissions raises pressing security and privacy challenges, leaving it vulnerable to spoofing, jamming, and eavesdropping [3]. Furthermore, the sheer volume of ADS-B signals demands advanced data processing techniques to ensure effective traffic management and the timely detection of anomalies.

Every ADS-B signal contains an 8 μs preamble and a 112 μs message, as shown in Fig. 1. It is primarily used to provide synchronization and frame detection at the receiver to interpret the message. Noticeably, a unique International Civil Aviation Organization (ICAO) address is embedded into each of the messages, identifying the source of the signal.

Fig. 1 ADS-B signal

There has been various research done in trying to overcome the security vulnerability of ADS-B signals as described by [4] and [5]. Spoofing involves recreating a fake signal that mimics the original or a fake aircraft to deceive the receiver, often causing confusion and inhibiting its ability to receive and report accurate information [3]. It is easy to carry out if the protocol is known. There are three main methods of spoofing:

A. *Transmitter sending Spoofed Signals*

The transmitter, ground or air, may send out falsified ADS-B signals to confuse the receiver. This ranges from modifying its message details to generating new signals from scratch. This creates the impression that there are many aircraft in the sky or that some aircraft are at a different location.

B. *Replay Attacks*

The transmitter may detect an actual signal from the air, then replay it sometime after the actual signal has passed. This is easy to conduct, making it dangerous. The spoofed signals received confuse the receiver, where an aircraft is declaring a different, past position from the actual one.

C. *Message Injection*

A transceiver can be built to detect and delete ADS-B signals from the aircraft, then replace them with a spoofed signal. Spoofed signals using this method can easily fool receivers, as the timing of signals received will correspond to the protocol. However, this is hard to create due to the fast-processing time that the transceiver must overcome.

While multiple techniques have been proposed against spoofing attacks, they require expensive array antennas and

979-8-3315-7206-8/25/$31.00 ©2025 IEEE

assume the ability for the receiver to receive the full signal [6]. When coupled with jamming and message deletion techniques, spoofing attacks pose a danger to the airspace.

This paper discusses the use of a small, labelled dataset of signal preambles to perform classification and hence verification of a signal source. The preamble forms the common denominator of all signals with the same protocol. It leverages on the slight differences in the electronics of the transmitter, such as the different radio frequency components, analogue circuits, onboard chips, or the slightest silicone difference of each chip manufactured. The electronics are viewed as a black box from the receiver's point of view.

Considering the potential threat of message injections, performing preamble analysis allows for the identification of the spoofed signal as well as the verification of the signal received. Preamble for classification is one of the least common methods. Typically, the full signal or the message is used as data to perform the machine learning. This is because the preamble carries little information as compared to the full signal or message itself. However, it is only at such low levels that the electronics black box differences can be seen without any bias. Performing analysis on such levels also requires receivers with a high sampling frequency of the analogue-to-digital converters (ADCs) to view the minute differences.

While there is much research on such machine learning based anti-spoofing measures on ADS-B signals, showing great success, there is limited research on anti-spoofing using the preamble of ADS-B signals. As such, the objective of this paper is to verify the source of the signal using only the preamble, classifying between coming from a particular source (target or true target), or other sources (non-target or false targets). Technically, we aim to perform spoofed signal detection based on the small-size data preamble using machine learning techniques such as k-Nearest Neighbors (kNN) and Multilayer Perceptron (MLP).

II. Related Work

Existing ADS-B anti-spoofing methods are briefly summarized as follows.

A. Received Signal Strength and Direction of Arrival

Using multiple antennae or an antenna array, the work in [7] discusses approaches using the received signal strength of the ADS-B messages as well as the direction of arrival (DoA) for location verification of the signal source. However, this is costly and cannot be replicated with a single antenna.

B. Doppler Shift Verification

As the aircraft travels between two points, the signals sent face the Doppler effect, distorting them slightly. This can be used in verifying the true source of the signal, and that the source is moving at a particular velocity with respect to the receiver [8]. However, this Doppler shift can still be spoofed and recreated by a malicious transmitter.

C. LSTM for Anomaly Detection

Long Short-Term Memory (LSTM) networks are a type of recurrent neural network (RNN) with the capability to learn and characterize time series data. Graves *et al.* [9] describe the bidirectional LSTM as being able to produce the highest accuracy of 70% for classification and achieving more than 90% precision in detecting spoofing attacks [10]. LSTM autoencoders were also used, showing better precision than standard autoencoders, as reported in [11]. However, this method assumes successful capture of signals and translates them into aircraft data. Message injections can make the aircraft slowly deviate from the flight path while passing the anomaly detection.

D. Fingerprinting

Due to the slight differences in the transponder's electronics, each transponder will have its own RF fingerprint as described in [12] and [13]. This can allow the receiver to know the identity of the sender without deciphering the message. As such, this is a powerful tool for anti-spoofing. In [14], the authors demonstrated the use of a Convolution Neural Network (CNN) to obtain an accuracy of 95% in aircraft classification. However, this method uses the full ADS-B signal received for anti-spoofing. Sources of message injection cannot be identified by this method. Therefore, we are motivated here to study source identification of deleted signals using message preambles.

III. Methodology

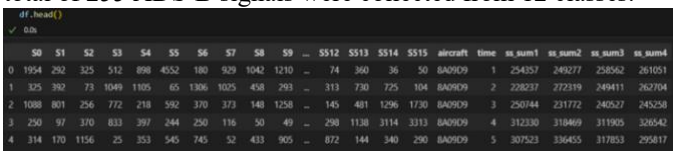
A. Data

To measure minute differences in a signal waveform above the noise, the signal must be sampled at a very high rate. As there are no such datasets available, we built a receiver using the ADRV9361-Z7035 system on module and collected 255 live ADS-B traces (data) in Singapore over a span of 1 hour.

The following assumptions and challenges were made in the collection:

- High noise environment
- Many signals were jammed or distorted
- Limited signals collected
- Multiple aircraft in the air

Live ADS-B signals were first collected by the receiver using the Integrated Logic Analyzer. Then, the signal source is identified and the signals are labelled. The preamble of each signal was extracted and compiled into a comma-separated values (CSV) file as shown in Fig. 2, with the label of the source (classes) and the sum of steady state values at each "high". A total of 255 ADS-B signals were collected from 12 classes.


df.head()
0.0s

	S0	S1	S2	S3	S4	S5	S6	S7	S8	S9	...	S512	S513	S514	S515	aircraft	time	ss_sum1	ss_sum2	ss_sum3	ss_sum4
0	1954	292	325	512	898	4552	180	929	1042	1210	...	74	360	36	50	8A09D9	1	254357	249277	258562	261051
1	325	392	73	1049	1105	65	1306	1025	458	293	...	313	730	725	104	8A09D9	2	228237	272319	249411	262704
2	1088	801	256	772	218	592	370	373	148	1258	...	145	481	1296	1730	8A09D9	3	250744	231772	240527	245258
3	250	97	370	833	397	244	250	116	50	49	...	298	1138	3114	3313	8A09D9	4	312330	318469	311905	326542
4	314	170	1156	25	353	545	745	52	433	905	...	872	144	340	290	8A09D9	5	307523	336455	317853	295817

Fig. 2 A snapshot of the preamble signal traces collected

The signals were then analyzed by plotting with the average signal trace for each class as presented in Fig. 3. It is noted that there are slight differences in the shape of the preamble signals, mainly on the steady state "high" value and the shape of the transition between "low" and "high".

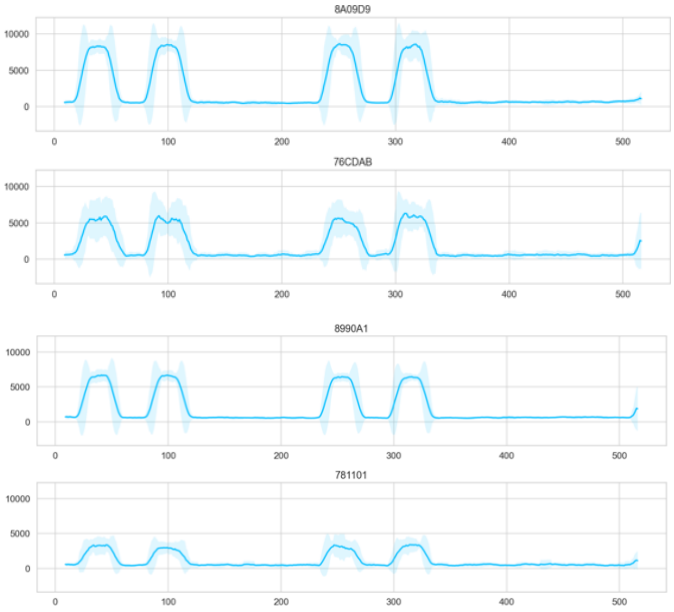


Fig. 3 Examples of plots from 4 different aircraft

B. Design

The signal features are first extracted using steady state amplitude analysis (SSAA) and transient state correlation analysis (TSCA), reducing the number of features from 516 to 12. With this process, a dataset of true targets and false targets can be generated. Next, kNN and MLP were used to classify between the true targets and false targets.

Each preamble has 4 "high" segments, denoted by $SS_i[t]$. As the power of the signal received is inversely proportional to the squared distance, and to accounting for a moving aircraft, linear extrapolation is used to estimate the amplitude, given by

$$SS_i'[t] = 2SS_i[t-1] - SS_i[t-2],$$

and an accuracy value, Δ_i, is determined for each "high" as

$$\Delta_i[t] = |SS_i'[t] - SS_i[t]|,$$

providing the scores for SSAA. Each preamble has 4 rise and 4 fall transitions, denoted by T_i. TSCA differentiates between the transitions in one preamble from another using Pearson correlations, ρ_i, on the target signal against the transitions of the averaged signal from the past 5 preambles of the target source.

C. k-Nearest Neighbors

Using Δ_i and ρ_i as inputs, a total of n inputs is created. kNN is an unsupervised learning algorithm, which is used to perform classification between target and non-target signals. The distances between the test data point, X, and the training data points, x_i, are computed and find its k nearest neighbors. The decision is based on the majority vote of the neighbors. The Minkowski distance generalizes the Euclidean distance (p=2) and the Manhattan distance (p=1), given by

$$d = (\sum_{i=1}^{n}(x_i - X)^p)^{\frac{1}{p}}.$$

D. Multilayer Perceptron Classifier

Here, Δ_i and ρ_i are used as inputs for the MLP classifier to classify between the target and non-target signals, as shown in Fig. 4. MLP is a supervised learning algorithm that uses multiple layers of neurons. Each neuron computes the weighted sum of the input, z, with a bias, b, using weights w on input x, i.e.,

$$z = b + \sum w_i x_i.$$

A non-linear activation function is then applied to the weighted sum. A forward propagation is used to compute the output y', and the loss L is computed against the actual output y using a binary cross-entropy loss function, i.e.,

$$L = -[ylog(y') + (1-y)\log(1-y')].$$

Backpropagation is then used to minimize the average loss by updating the weights and biases. The process is optimized using an Adam optimizer to iteratively refine the weights and biases.

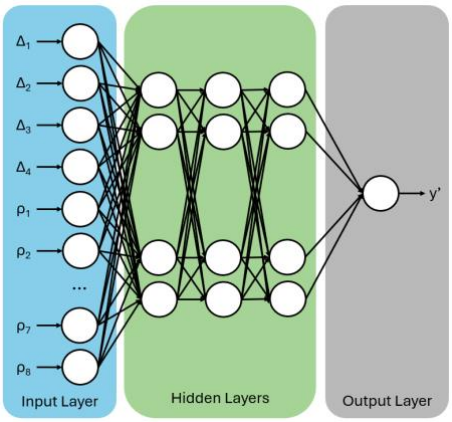


Fig. 4 Multilayer Perceptron

IV. Results and Discussion

As an anti-spoofing technique on preamble signals, the aim is to identify if the signal is from the target aircraft, hence, precision is preferred over accuracy. Precision is computed as

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}.$$

kNN is applied on the results of SSAA and TSCA, Δ_i and ρ_i, to detect targets and non-targets without a manual threshold input. A test size of 15% and Euclidean distance was used.

To determine the best value of k, the elbow method for distortion is plotted in Fig. 5, which calculates the average squared distance between each point and the centroids. The

result of the analysis is shown in Table I by varying the number of nearest neighbours *k*.

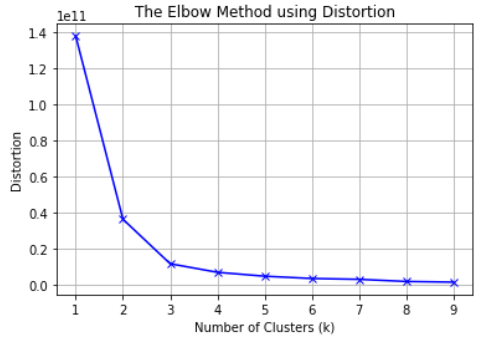


Fig. 5 Elbow method using distortion

Table I: Precisions of kNN

Test	Precision	
k	p=1	p=2
1	58.62	56.52
2	58.62	56.52
3	62.50	74.07
4	58.62	72.73
5	53.85	61.11
6	53.85	66.67
7	56.00	65.00
8	57.14	66.67
9	59.26	70.00
10	59.26	70.00
11	62.96	78.95
12	62.07	72.22
13	65.38	76.47
14	60.71	71.43
15	64.29	81.25
16	62.07	76.92
17	60.87	72.73
18	60.00	75.00
19	58.33	75.00
20	62.96	66.67

Table II: Higher precisions using MLP

Hidden Layer 1	Hidden Layer 2	Hidden Layer 3	Dataset 1 Precision	Dataset 2 Precision	Dataset 3 Precision
4	5	5	100	91.67	90.91
4	10	3	100	93.33	100
10	2	18	100	90.91	90.91
10	6	13	100	100	93.75
14	2	15	100	92.86	81.48
14	2	19	92.59	100	100

The results for kNN are limited by the data size and the cleanliness of the data. As such, for noisy data, the precision is considered to be exceptionally good. For small data sets, it is recommended to use the elbow method with k=3 and p=2 as the test data set will be too small to provide an accurate result.

A three hidden layered MLP model was built, varying the number of ReLU cells in each layer. The precision values of the test results are summarized in Table II, and the results of higher precision, more than 90%, are noted. To ensure that the model works for other data sets, it was trained with 2 other datasets, generated using the results from SSAA and TSCA.

The hidden layers of (4, 5, 5), (4, 10,3), (10, 2, 18), (10, 6 ,13), and (14,2,19) gave a constant high precision result. The average precision and accuracy were computed Table III.

Table III: Average accuracy and precision of MLP technique

Hidden Layer 1	Hidden Layer 2	Hidden Layer 3	Average Precision	Average Accuracy
4	5	5	94.19	51.52
4	10	3	97.78	54.54
10	2	18	93.94	52.73
10	6	13	97.92	58.18
14	2	19	97.53	64.24

According to the results, (14, 2, 19) is recommended as the average precision remained high at 97% while having a higher average accuracy of 64% (see Fig. 6).

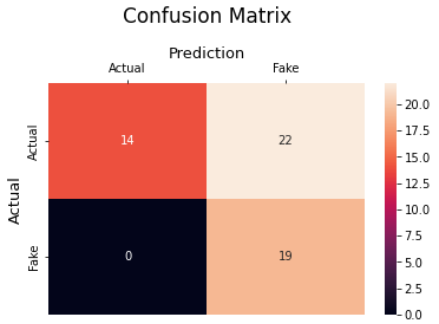


Fig. 6 Confusion matrix of MLP on dataset 2

V. CONCLUSION

This is the first paper that exploits preamble signals for ADS-B anti-spoofing. As shown in this paper, preamble analysis for verification of an ADS-B signal source is a challenging task, as the signals obtained are generally contaminated by strong noises and would require a receiver with a high sampling frequency. However, we have demonstrated that it provides strong anti-spoofing capability even against complex injection techniques. As such, we have curated our own dataset and developed a kNN and a MLP based classifier, respectively. Noticeably, the MLP models have shown better potential by the kNN classifiers, obtaining a precision of 97% with an accuracy of more than 60% with an appropriate selection of structure.

A precision of 99% is required for user confidence that the identified aircraft is what it was predicted to be. A wrong identification of aircraft can lead to major air traffic accidents. Hence, an extremely strict precision is required. Future works can explore further optimization of the learning models and performing classification using larger datasets. Extending the idea in this paper, one may use a similar approach in filtering out spoofed signals received. In this case, the recall metrics can be used to ensure that all spoofed targets are removed.

References

[1] Federal Aviation Administration, "NextGen - SESAR State of Harmonisation Third edition," Publications Office of the European Union, Luxembourg, Sep. 2018. Accessed: Mar. 11, 2025. [Online]. Available: https://www.faa.gov/sites/faa.gov/files/2022-06/NextGen-SESAR_State_of_Harmonisation.pdf

[2] Flight Safety Foundation, "Benefits Analysis of Space-Based ADS-B," Jun. 2016. Accessed: Mar. 11, 2025. [Online]. Available: https://flightsafety.org/wp-content/uploads/2016/10/ADS-B-report-June-2016-1.pdf

[3] S. Khandker, H. Turtiainen, A. Costin, and T. Hamalainen, "Cybersecurity Attacks on Software Logic and Error Handling within ADS-B implementations: Systematic Testing of Resilience and Countermeasures," IEEE Transactions on Aerospace and Electronic Systems, vol. 58, no. 4, pp. 2702–2719, 2021, doi: https://doi.org/10.1109/taes.2021.3139559.

[4] C. Shi-yi and L. Yu-bai, "Error Correcting Cyclic Redundancy Checks based on Confidence Declaration," 2006 6th International Conference on ITS Telecommunications, pp. 511–514, Jun. 2006, doi: https://doi.org/10.1109/itst.2006.288954.

[5] A. G. Aydn, M. Y. Polat, and A. Öncü, "ADS-B-Feature-Based Error Correction Method Implementation and Validation Using SDR," IEEE Transactions on Aerospace and Electronic Systems, vol. 60, no. 2, pp. 1482–1489, Nov. 2023, doi: https://doi.org/10.1109/taes.2023.3337211.

[6] Z. Wu, T. Shang, and A. Guo, "Security Issues in Automatic Dependent Surveillance - Broadcast (ADS-B): A Survey," IEEE Access, vol. 8, pp. 122147–122167, 2020, doi: https://doi.org/10.1109/ACCESS.2020.3007182.

[7] S. Rudys, J. Aleksandravicius, R. Aleksiejunas, A. Konovaltsev, C. Zhu, and L. Greda, "Physical Layer Protection for ADS-B against Spoofing and Jamming," International Journal of Critical Infrastructure Protection, vol. 38, p. 100555, Sep. 2022, doi: https://doi.org/10.1016/j.ijcip.2022.100555.

[8] N. Ghose and L. Lazos, "Verifying ADS-B Navigation Information through Doppler Shift Measurements," in 2015 IEEE/AIAA 34th Digital Avionics Systems Conference (DASC), Sep. 2015, pp. 1–27. doi: https://doi.org/10.1109/dasc.2015.7311578.

[9] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," Neural Networks, vol. 18, pp. 602–610, Jul. 2005, doi: https://doi.org/10.1016/j.neunet.2005.06.042.

[10] J. Wang, Y. Zou, and J. Ding, "ADS-B Spoofing Attack Detection Method Based on LSTM," EURASIP Journal on Wireless Communications and Networking, vol. 2020, Aug. 2020, doi: https://doi.org/10.1186/s13638-020-01756-8.

[11] A. Fried and M. Last, "Facing Airborne Attacks on ADS-B Data with Autoencoders," Computers & Security, vol. 109, p. 102405, Oct. 2021, doi: https://doi.org/10.1016/j.cose.2021.102405.

[12] A. Jagannath, J. Jagannath, and P. S. P. V. Kumar, "A Comprehensive Survey on Radio Frequency (RF) fingerprinting: Traditional approaches, Deep learning, and Open Challenges," Computer Networks, vol. 219, p. 109455, Dec. 2022, doi: https://doi.org/10.1016/j.comnet.2022.109455.

[13] N. Soltanieh, Y. Norouzi, Y. Yang, and N. C. Karmakar, "A Review of Radio Frequency Fingerprinting Techniques," IEEE Journal of Radio Frequency Identification, vol. 4, no. 3, pp. 222–233, 2020, doi: https://doi.org/10.1109/jrfid.2020.2968369.

[14] Haoran Zha, Q. Tian, and Y. Lin, "Real-World ADS-B signal recognition based on Radio Frequency Fingerprinting," in 2020 IEEE 28th International Conference on Network Protocols (ICNP), Oct. 2020. doi: https://doi.org/10.1109/icnp49622.2020.9259404.

[15] J. Habibi Markani, A. Amrhar, J.-M. Gagné, and R. J. Landry, "Security Establishment in ADS-B by Format-Preserving Encryption and Blockchain Schemes," Applied Sciences, vol. 13, no. 5, p. 3105, Jan. 2023, doi: https://doi.org/10.3390/app13053105.