Synchronization-Free GPS Spoofing Detection with Crowdsourced Air Traffic Control Data

Gaoyang Liu[1], Rui Zhang[2], Chen Wang[1], Ling Liu[3]
[1]*Huazhong University of Science and Technology, Wuhan China, 430074*, {liugaoyang, chenwang}@hust.edu.cn
[2]*Wuhan University of Technology, Wuhan China, 430070*, zhangrui@whut.edu.cn
[3]*Georgia Institute of Technology, Atlanta, GA USA 30332*, lingliu@cc.gatech.edu

***Abstract*—GPS-dependent localization, navigation and air traffic control (ATC) applications have had a significant impact on the modern aviation industry. However, the lack of encryption and authentication makes GPS vulnerable to spoofing attacks with the purpose of hijacking aerial vehicles or threatening air safety. In this paper, we propose GPS-Probe, a GPS spoofing detection algorithm that leverages the ATC messages that are periodically broadcasted by aerial vehicles. By continuously analyzing the received signal strength indicator (RSSI) and the timestamps at server (TSS) of the ATC messages, which are monitored by multiple ground sensors, GPS-Probe constructs a machine learning enabled framework to estimate the real position of the target aerial vehicle and to detect whether or not the position data is compromised by GPS spoofing attacks. Unlike existing techniques, GPS-Probe neither requires any updates of the GPS infrastructure nor updates of the GPS receivers. More importantly, it releases the requirement on time synchronization of the ground sensors distributed around the world. Using the real-world ATC data crowdsourced by the OpenSky Network, our experiment results show that GPS-Probe can achieve the detection accuracy and precision, of 81.7% and 85.3% respectively on average, and up to 89.7% and 91.5% respectively at the best.**

***Index Terms*—GPS spoofing attacks, air traffic control data, machine learning, OpenSky Network.**

I. Introduction

In recent years, the Global Localization System (GPS) has become a ubiquitous source of location, tracking, and navigation information for more than a billion devices [1], [2]. Among the fields of GPS deployments, the aviation industry is one of the earliest civil areas, and its reliance on GPS increases rapidly due to the requirement of navigation and air traffic monitoring. With the dramatic growth of Unmanned Aerial Vehicles (UAVs), GPS is mission-critical not only for airplanes but also for UAVs, ranging from consumer-class drones to tactical and strategic UAVs.

Unfortunately, the civilian GPS signals sent by the satellites are neither authenticated nor encrypted [3], [4]. A malicious transmitter can easily imitate the legitimate GPS signals and emit fake signals with a higher power or slightly different time delays. Aerial vehicles' GPS receivers are vulnerable to these fake signals since the receivers often hook the signal with a higher strength. As a consequence, the aerial vehicle's position calculated with the spoofed signals can be tampered by the attackers arbitrarily [5]–[7]. One best-known real-world case happened in 2011, when a CIA stealth UAV (RQ-170) was captured by Iranians, who hijacked its GPS coordinates and safely brought it down[1].

Driven by the increasing threats, a number of countermeasures have been developed, which generally fall into two class. The first class aims at authenticating signals from satellites by adding extra signals that are unpredictable to attackers [8]–[10]. However, this type of protection requires a hardware upgrade of the GPS infrastructure and can be cracked easily by replay attacks. The second class focuses on the spoofing detection at signal-processing level, which utilize special-designed GPS receivers with adapted signal processing techniques and enhanced electronic circuits. With customized receivers, spoofing attacks can be detected from more relevant information, such as the angle-of-arrival of GPS signals [11], random antenna motion [12], [13], and automatic gain control on the radio frontend [14]. Nevertheless, most if not all existing countermeasures require far-reaching modifications of either the GPS infrastructure or the receiving devices, yielding them unlikely to be deployed in the near future.

In response to providing short-term practical solutions, the latest work dubbed Crowd-GPS-Sec proposed to leverage crowdsourcing to monitor the air traffic control (ATC) messages periodically broadcasted by aerial vehicles [15]. Crowd-GPS-Sec first estimates the position of the target aerial vehicle based on the time difference of arrival (TDoA) of the ATC messages received by different ground sensors. By testing out the inconsistencies between the estimated position and the GPS-derived position contained in the ATC messages, GPS spoofing attacks can thus be detected without the need to update neither the GPS infrastructure nor the GPS receivers. One major challenge that hinders the application of Crowd-GPS-Sec, however, is its high reliance on precise time synchronization of the ground sensors, which is also the inherent limitation of TDoA-based localization technologies [16]. As the ATC messages are spread approximately $300,000km/h$, a small time offset will lead to a large localization error, which

This work was supported in part by the National Natural Science Foundation of China under Grants 61872416, 51879210, 61671216, 61871436, 51479159, 61872415 and 61702204; by the Fundamental Research Funds for the Central Universities of China under Grant 2019kfyXJJS017; and by the fund of Hubei Key Laboratory of Transportation Internet of Things under Grant 2018IOT004. Ling Liu's research is partially support by the National Science Foundation under NSF Grants 1547102, 1564097 and an IBM faculty award. The corresponding author of this paper is Chen Wang.

[1]https://en.wikipedia.org/wiki/Iran-U.S._RQ-170_incident

2375-0324/19/$31.00 ©2019 IEEE
DOI 10.1109/MDM.2019.00054

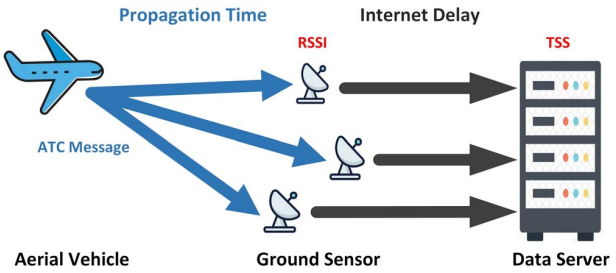


Fig. 1. Schematic of the crowdsourced ATC data transmission.

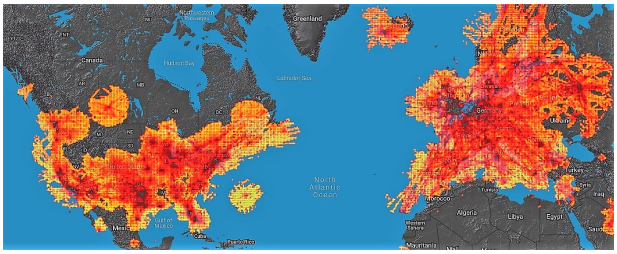


Fig. 2. Worldwide coverage of the OpenSky Network.

in turn degrades the attack detection accuracy on the whole.

In this paper, we propose GPS-Probe, a synchronization-free GPS spoofing detection scheme leveraging crowdsourced ATC data. Instead of using TDoA, GPS-Probe leverages the *received signal strength indicator (RSSI)* monitored by each ground sensor and the *timestamps at server (TSS)* as features (cf. Fig. 1), and relocates the target aerial vehicles via machine learning techniques. The estimated position is further utilized to detect the GPS spoofing attacks by analyzing its difference from the GPS-derived position. Specifically, we have made the following major contributions:

- We propose GPS-Probe, the first work to take advantage of RSSI and TSS features for GPS spoofing detection with crowdsourced ATC data, which fundamentally changes standard TDoA-based technique in the sense of releasing the time synchronization requirement.
- We propose an adaptive k nearest neighbors (k-NN) classifier, aiming to suppress the noise in RSSI and to mitigate the errors of TSS-based localization, which further helps improve the attack detection performance.
- We evaluate GPS-Probe with real-world data from the OpenSky Network[2] [17] which maintains a network of more than 800 ground sensors around the world (cf. Fig. 2). The results show that GPS-Probe is able to detect GPS spoofing attacks globally in less than 5 seconds, and the detection accuracy can achieve 90% with attack range of $10km$.

The remainder of this paper is organized as follows. In Section II, we provide some preliminary knowledge. In Section III, we present more details on the design of GPS-Probe. Section IV evaluates GPS-Probe with real-world data from the OpenSky Network. Section V summarizes some related work and finally Section VI concludes the paper.

II. Preliminary

A. GPS and GPS Spoofing Attacks

GPS [1] is a satellite-based navigation network of 32 satellites located in the medium Earth orbit, and these satellites carry very stable atomic clocks that are synchronized with one another and with the ground clocks. GPS provides geo-location, velocity and time information to GPS receivers anywhere on or near the Earth where there is an unobstructed line of sight to four or more GPS satellites. By measuring the time of arrival from unobstructed satellites, GPS receivers can calculate the distances to each satellite, and then derive the position and the local time of the receivers.

GPS is typically used by pilots or UAVs' controllers for self localization, as well as remote air-traffic surveillance and collision avoidance applications. For the safety of air traffic, the aerial vehicles need to periodically broadcast position, heading and velocity advertisements to notify neighboring aerial vehicle and ground sensors by the Automatic Dependent Surveillance-Broadcast (ADS-B) system[3] or the Flarm[4] system.

Although GPS is widely deployed in civilian aviation, the lack of encryption and authentication makes it vulnerable for spoofing attacks [3], [4]. Attackers can easily mitigate the legitimate signals of GPS satellites, modify the data of GPS signals, and transmit these deliberately crafted signals to the target location with a higher power. Whenever a GPS receiver locks on to the spoofed signals, the position advertisements of aerial vehicles derived from GPS can be tampered by attackers arbitrarily [5]–[7].

B. Crowdsourced ATC Data

ATC [18] is a service provided by ground controllers who direct aerial vehicles passing through controlled airspace on the ground. For the purpose of collision avoidance and air traffic organization, the aerial vehicles periodically broadcast the ADS-B or Flarm messages to declare their flight status. These status messages received by ATC ground stations construct the so-called ATC data. Note that as the ADS-B and Flarm communication protocols are open-source to public, anyone can monitor and collect these ATC data with state-of-art soft defined radio devices.

The OpenSky Network [17] adopted in this paper is a crowdsourcing initiative to collect ATC data and making the data available to the public. The ground sensors of OpenSky are most installed and operated by aviation enthusiasts and volunteers. The volunteers continuously monitor the ATC data, which are then sent to the data center through Internet. As of this writing, it collects more than $200,000$ messages per second at peak time from over 800 ground sensors distributed

[2]https://opensky-network.org/

[3]http://www.ads-b.com/

[4]https://flarm.com/

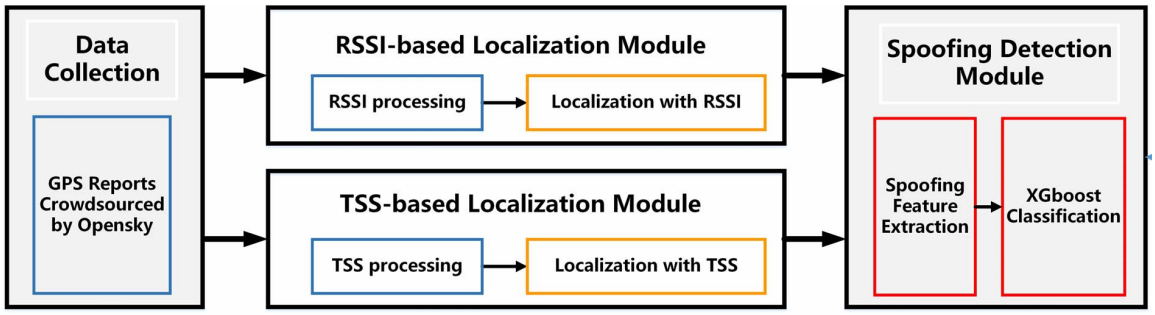


Fig. 3. System overview of GPS-Probe.

all over the world (cf. Fig. 2). The coverage in all regions is constantly growing, yet a single sensor can receive GPS signals up to a distance of $700km$, allowing to cover the whole world with a few thousand ground sensors [19].

C. RSSI

Available in mainstream wireless signal measurements, RSSI represents the power of a received radio signal after the propagation attenuation, and is widely used in radio frequency (RF) system to provide necessary information to adjust receiver antenna gain. RSSI of received signals is calculated as follows:

$$RSSI_{RF} = 10\log(P_r/P_t), \tag{1}$$

where P_r (resp. P_t) is the power of transmitted (resp. received) signal. In the free-space model which is widely used in line-of-sight RF localization [20], the following relationship holds:

$$P_r \propto P_t \cdot (\frac{1}{d})^2, \tag{2}$$

where d is the distance between the RF transmitter and receiver. From Eq. (1) and (2), we can observe that RSSI is correlated with the propagation distance. The shorter the propagation distance is, the higher RSSI value will be.

As for the OpenSky Network, the sources of the RF signals are the aerial vehicles which periodically broadcast the ATC messages, while the ground sensors serve as the RF receivers. The transmitting power of the ATC messages is determined by the electronic circuits on the aerial vehicles, and it is impossible to disturb the RF transmission modules by existing GPS spoofing attacks. After propagation, the ATC messages are received by the ground sensors of the OpenSky Network, and the RSSI of the received signals are credible even the GPS information is modified by malicious attackers. As such, we can calculate the real geoposition of the aerial vehicles by leveraging the distance information hidden behind the RSSI.

III. GPS-Probe Design

In this paper, we propose GPS-Probe as an independent infrastructure on the ground that analyzes the ATC messages and detects the GPS spoofing attacks continuously. To this end, GPS-Probe mainly involves the following three modules (cf. Fig. 3):

(1) **RSSI-based Localization Module.** In this module, GPS-Probe leverages RSSI of ATC messages to estimate the position of target aerial vehicle. We first remove the outliers and noises of the RSSI measurements, and utilize the processed RSSI to construct a RSSI fingerprint map. Next, we divide the surveillance area into $M \times N$ squares and construct the RSSI feature vectors corresponding to each square by combining the original and processed RSSI. Then we train a classification model with these RSSI feature vectors and their corresponding square labels. After determining which square the target aerial vehicle is in, GPS-Probe can obtain the location estimation Loc_R by our adaptive k-NN classifier.

(2) **TSS-based Localization Module.** In this module, we leverage the TSS values of ATC messages to estimate the position of the target aerial vehicles. We first remove the Internet delays in TSS measurements, which is caused by the ground sensors transmitting ATC messages to the data center. Then we construct TSS feature vectors corresponding every square in the surveillance area. After the preprocessing of TSS measurements, we take a similar approach to the RSSI-based localization module to obtain the position estimation Loc_T.

(3) **GPS Spoofing Detection Module.** Given the estimated positions Loc_R and Loc_T, GPS-Probe detects whether or not the target aerial vehicle is attacked by GPS spoofing. To achieve this, we leverage Xgboost to train a binary classification detection model on the positions derived from our localization modules and the ATC messages. Finally for any aerial vehicle, GPS-Probe monitors its RSSI and TSS of received GPS RF signals at different ground sensors in real time and outputs the detection result.

A. RSSI-based Localization

1) RSSI Preprocessing: RSSI measurements of ATC messages obtained from ground sensors contain outliers and noises from various sources, such as multipath fading, transmission power adaptation at the sender, and random amplitude errors. To remove the outliers, we simply utilize the Hampel identifier [21]. For denoising, we apply the classical yet simple sliding window averaging method, with a window width of 30 seconds on the raw RSSI series of one specific aerial vehicle. The RSSI preprocessing enables us to remove the outliers and

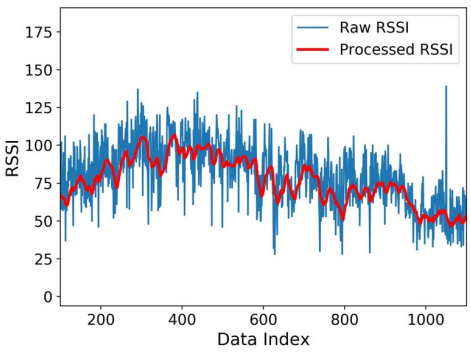


Fig. 4. RSSI Preprocessing.

noise while keeping the effective information of RSSI (cf. Fig. 4).

2) Feature Extraction: The key of our localization approach is making full use of the processed RSSI measurements. According to Eq. (1) and (2), we can derive the following relationship:

$$10\exp(-RSSI/20) \propto d, \tag{3}$$

where d is the distance from the aerial vehicle to the ground sensor, and $RSSI$ is the signal strength indicator after preprocessing.

To improve the localization accuracy, we construct a new feature which is linearly dependent on the distance d as follows:

$$RSSI_{+} = 10\exp(-RSSI/20). \tag{4}$$

$RSSI_{+}$ amplifies $RSSI$ exponentially and thus can provide more distinction among adjacent locations, which can benefit our machine learning model for more accurate localization. We thus combine $RSSI$ and $RSSI_{+}$ together as the feature vector $RSSV$ for machine learning algorithms to estimate the position of a target aerial vehicle (cf. Table I).

3) Localization with RSSI: With $RSSV$, we adopt classic k-nearest neighbors (k-NN) classifier for localization, which contains an offline training phase and an online localization phase. In the offline phase, we build a 2D grid of $M \times N$ squares in the surveillance area as in [22], and construct RSSI fingerprint map, i.e., the RSSI feature vectors and the corresponding grid labels among the ground sensors for every position.

In the online localization phase, the RSSI feature vector $RSSV_{inco}$ of an incoming ATC message from the target aerial vehicle is calculated at first. Using the standard k-NN classifier, we can obtain the closest square that matches $RSSV_{inco}$ as:

$$\begin{aligned} \arg\min_{(m,n)} \ & \frac{1}{I^R_{m,n}} \sum_{i=1}^{I^R_{m,n}} \left\| RSSV_{inco} - RSSV^i_{(m,n)} \right\|_2, \\ & m \in [1, 2 \cdots M], n \in [1, 2 \cdots N], \end{aligned} \tag{5}$$

where $RSSV^i_{(m,n)}$ is the i_{th} ($\in I^R_{(m,n)}$) RSSI feature vector of the training data which locates in the square (m, n). By doing so, we can narrow down the target location into a particular square (cf. the left part of Fig. 5).

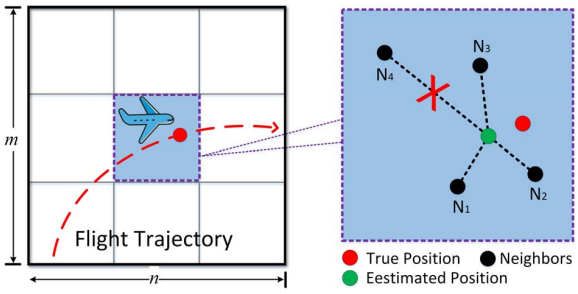


Fig. 5. A toy model of k-NN localization.

In what follows, we focus on calculating the fine-grained estimation of the target location in the square. Adopting the standard k-NN classifier in the square seems like a reasonable method. However, the accuracy is influenced by the value of k largely due to a relatively small number of training data in the square. As the toy model shown in the right part of Fig. 5, if we select $k = 4$ neighbors, the outlier N_4 will be included to obtain an estimated position which highly deviates from the true position. Likewise, estimating the position by averaging $k = 2$ neighbors also results in a relatively large error since the eligible point N_3 is wrongly treated.

We thereby introduce an adaptive k-NN approach based on the gradient of Euclidean distance series between $RSSV_{inco}$ and $RSSV^i_{(m,n)}$:

$$G_R = \nabla_{i \in I^R_{(m,n)}} \ Sort \ D^i_R, \tag{6}$$

where $D^i_R = \left\| RSSV_{inco} - RSSV^i_{(m,n)} \right\|_2$, and $Sort$ means sorting the distance series in ascending order. To choose a suitable value of k, we traverse the gradient series G_R to find the first local maximum and its corresponding serial number K_{dema}. We experimentally find that K_{dema} serves as a demarcation point between the proper and improper value of k (cf. Fig. 6). After obtaining the proper number of selected neighbors, the final output of the RSSI-based localization module, i.e., Loc_R, can be obtained by the standard k-NN classifier

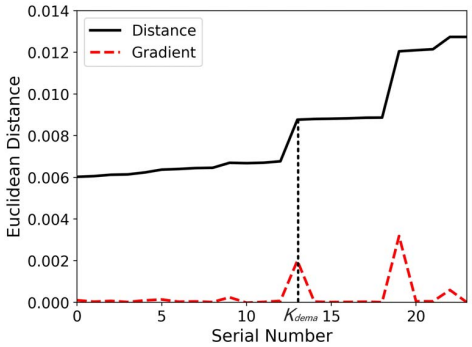


Fig. 6. Sorted distance series and corresponding gradient series.

TABLE I
FEATURE VECTORS, MACHINE LEARNING ALGORITHMS AND OUTPUTS OF THREE MODULES.

Module	Feature Vector	Machine Learning Algorithm	Output
RSSI Localization	$RSSV = [RSSI,\ RSSI_+]$	k-NN + adaptive k-NN	Loc_R
TSS Localization	$TSSV = [TSS,\ TSS_{diff},\ SenMarker]$	k-NN + adaptive k-NN	Loc_T
Spoofing Detection	$Det_{FV} = [Loc_R - Loc_{GPS},\ Loc_T - Loc_{GPS}]$	Xgboost	Spoofed or Not

in the straightforward manner. With the adaptive k-NN, we can thus remove the outliers from the nearest neighbors while picking out appropriate neighbors from $RSSV_{inco}$.

B. TSS-based Localization

In the data obtained from OpenSky Network, we find that approximately a quarter of the RSSI measurements of ATC messages are missing, rendering relatively low precision of the RSSI-based localization. To improve the localization accuracy, we also implement a supportive TSS-based localization module in this section.

1) TSS Preprocessing: TSS measurements provided by OpenSky Live API contain two parts (cf. Fig. 1): one is the propagating time caused by GPS RF signals transmitting from the aerial vehicle to the ground sensor, and the other is the Internet delay introduced by the data transmission from the ground sensor to the OpenSky data server. Generally, the propagating time is tens of microseconds while the Internet delay is tens of milliseconds. Thus the Internet delay overrides the fluctuation of propagation time caused by the distance changes between the aerial vehicle and the ground sensor.

For aerial localization with the time measurements of TSS, we need to calculate the ATC messages' propagation times through subtracting the transmitting time of the aerial vehicles, and then remove the Internet delays. The transmitting times of ATC messages can be easily obtained by OpenSky Network. The distribution of Internet delay approximately obeys the Weibull distribution [23]. We remove the expectation, or the mean value of the Internet delay from TSS measurements. This preprocessing method enables us to provide our localization model with the TSS measurements which are relatively close to the transmitting time of ATC messages and thus can reflect the real distance between the signal sources and the ground sensors.

2) Feature Extraction: As laid out above subsection, TSS measurements reflect the real d between the signal sources and the ground sensors. However, it is hard to get the precise transmitting time from the GPS transmitter to the ground sensor even we remove the Internet delays in advance. In our TSS-based localization module, we construct several new features to improve the localization performance. At first, we make use of the difference values between any two TSS measurements among different ground sensors as the additional features of different locations:

$$\begin{aligned} TSS_{diff} = [T_{1,1}, T_{1,2}, \cdots, T_{i,j}, \cdots, T_{Q,Q}], \\ i,j \in [1,2,\cdots Q],\ Q \in [1,2,\cdots,P], \end{aligned} \tag{7}$$

where $T_{i,j} = T_i - T_j$ is the difference value of TSS measurements obtained at i_{th} and j_{th} receiving ground sensors. P is the total number of the ground sensors and Q is the number of ground sensors which receive the ATC messages. TSS_{diff} implies the order in which the same message arrives at different ground sensors, and the order can represent the relative position of the aerial vehicle to the receiving ground sensors. $T_{i,j}$ represents the relative position to the ground sensors more specifically.

Besides, since the monitoring range of the ground sensor is limited, which sensors receive the ATC message is a useful feature for localization. Thus we construct a new feature vector $SenMarker$, which marks the receiving sensors of each ATC message. $SenMarker$ is a $P \times 1$ vector. If an ATC message is received by p_{th} sensor, we will set the corresponding value in $SenMarker$ to 1, while other values remain 0.

We combine the TSS measurements, TSS_{diff} and $SenMarker$ together as the feature vector $TSSV$ (cf. Table I) and use $TSSV$ as the fingerprints of the aerial vehicle positions derived from the unassailed ATC messages. Then we leverage these fingerprints to estimate the geometric coordinates of a target aerial vehicle.

3) Localization with TSS: With $TSSV$, we use the same process as that of RSSI-based localization. In the offline training phase, we construct the TSS fingerprint map in the 2D grid of $M \times N$ squares. In the online localization phase, we first train a k-NN classifier to pinpoint the target location into a particular square:

$$\begin{aligned} \underset{(m,n)}{\arg\min}\ \frac{1}{I^T_{m,n}} \sum_{i=1}^{I^T_{m,n}} \left\| TSSV_{inco} - TSSV^i_{(m,n)} \right\|_2, \\ m \in [1,2\cdots M], n \in [1,2\cdots N], \end{aligned} \tag{8}$$

where $TSSV_{inco}$ is the TSS feature vector of the incoming ATC message, and $TSSV^i_{(m,n)}$ is the i_{th} ($\in I^T_{(m,n)}$) TSS feature vector of the training data which locates in the square (m,n). The final output of the TSS-based localization module, i.e., Loc_T, is then obtained by the adaptive k-NN classifier in a similar way as in RSSI-based localization module.

C. GSP Spoofing Detection

After obtaining the estimated locations (i.e., Loc_R and Loc_T), we can focus on detecting the spoofing attacks, based on the deviation between the estimated positions and the GPS-derived position contained in the ATC messages. Since the attackers have the ability to manipulate the GPS-derived positions of aerial vehicles arbitrarily, the position offset modes

can vary a lot. Thus using the deviation between the estimated and true positions directly cannot achieve high detection accuracies. To address this issue, we propose the following algorithm based on Xgboost [24], a reliable technique which has the capability to learn various position deviation modes and even to resist the impact of localization errors.

1) Spoofing Feature Extraction: As is known that proper input features containing abundant information could improve the performance of Xgboost classifier dramatically. In order to achieve a high detection accuracy, we thus reveal the latent differences between spoofed locations and real locations of aerial vehicles to construct the input classification features. We calculate the latitude and longitude differences between the GPS-derived position and Loc_R (resp. Loc_T), and take them as a part of the detection feature vector Det_{FV} (cf. Table I).

2) Spoofing Detection based on Xgboost: With Det_{FV}, GPS-Probe trains an Xgboost classifier to infer whether an aerial vehicle is spoofed or not. Thus spoofing detection is transformed into a binary classification problem. In order to achieve a high detection accuracy, we need to adjust a set of parameters of Xgboost such as *max-depth*, *num-trees*, *learning-rate*. Empirically, a large value of *max-depth* or *num-trees* can result in a better performance of our detection model, whereas the complexity and the training time of the model would increase at the same time. In addition, a too complex model will degrade its ability of generalization on an unmet set of data. Thus, a suitable set of model parameters can maintain a good performance while avoiding overfitting or training for a long time.

In practical, we leverage the function *GridSearchCV* of scikit-learn machine learning library[5] to find the proper parameter values. We use standard accuracy metrics on training and testing dataset for evaluating different parameters. When the detection accuracy of the model in the training set does not increase, or the accuracy in the testing set starts to decline, *GridSearchCV* stops searching and outputs the current parameters as the optimal selection.

After tuning the parameters of our spoofing detection model, we obtain the binary model with two output classes, "Spoofed" and "Not Spoofed". With the constructed feature vector of an incoming ATC message, GPS-Probe tests Det_{FV} through the binary model and can finally get the detection results of GPS spoofing attacks.

IV. Performance Evaluation

To evaluate the feasibility and effectiveness of GPS-Probe in terms of spoofing detection accuracy and detection delay, we applied GPS-Probe to real-world ATC data from the OpenSky Network and built a simulation framework to generate impacts of spoofing attacks on GPS coordinates. We further compare GPS-Probe against Crowd-GPS-Sec [15] to show its superiority.

[5]https://scikit-learn.org/stable/modules/grid_search.html

A. Setup

The ATC data we used to evaluate GPS-Probe is from the OpenSky Network, which contains $274,583$ ATC messages of 181 aerial vehicles over a period of $1h$. Each ATC message has been received by at least 3 ground sensors, providing us with abundant RSSI and TSS measurements to estimate the target position. In addition, in the offline training process, all ATC messages are marked with pre-defined labels of grid squares, and are presented in terms of latitude and longitude. The data set is divided into the following three parts:

- Training set, which is used to build the RSSI/TSS localization fingerprints and the spoofing detection module.
- Development set, which is used to help us adjusting the module parameters to achieve a better detection accuracy.
- Testing set, which is leveraged to evaluate the performance of GPS-Probe.

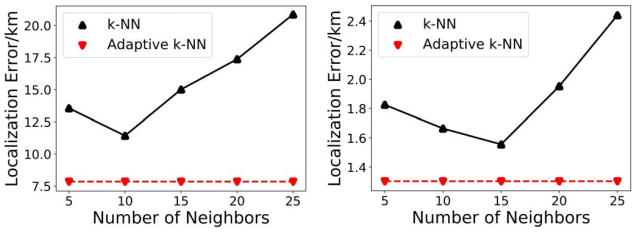
Simulation Framework. As the ATC data crowdsourced by OpenSky Network contains GPS-derived position reports without being spoofed, we build a simulation framework to imitate the results of already spoofed coordinates of GPS-derived position. It is noticed that even the same method of GPS spoofing may result in different localization deviations in both directions and distances. So we add random noise to the latitude and the longitude of the selected ATC messages. Subsequently, we put the spoofed data back while ensuring that the proportion of spoofed data is consistent in the training set, the development set and the testing set. Note that for each simulation, we performed 100 randomized runs to average out the randomness.

Grid Design. Since the ATC data we used are generated by airliners and the civil aircrafts usually cruise at an altitude of $10km$ except for take-off and landing phases, we construct a 2D grid over a typical flight altitude of $10km$ with a size of 5.5 degree longitude and 2.5 degree of latitude, which covers an area of $610km \times 200km = 122,000km^2$. $122,000km^2$ are typical for wide area ATC which includes a sufficient number of OpenSky ground sensors. The grid squares are evenly spaced in the monitoring area and the number of squares is a trade-off between computation time and performance. Specifically, we separate the monitored aerial area into $1,375$ parts with an interval of 0.1 degree.

B. Performance of Localization

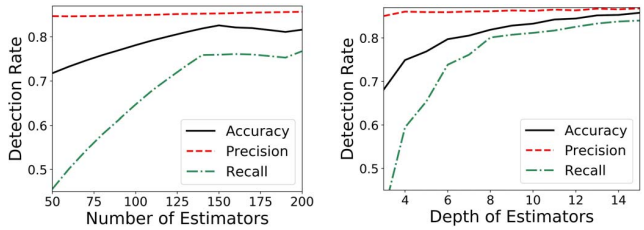
In this section, we evaluate the location accuracy of the RSSI and TSS based grid localizations, and further compare our adaptive k-NN against the standard k-NN method for the fine-grained localization within a certain square.

Localization Based on RSSI. It is noticed that the nearest neighbors are selected from the dataset within one particular square, thus grid classification has enormous impacts on the localization accuracy. We first make use of classic k-NN algorithm to mark the label of the square where the incoming ATC message lies in. In our experiments we find that the prediction accuracy decreases dramatically with the value k when $k < 100$. Furthermore, the accuracy gain



(a) RSSI-based Localization (b) TSS-based Localization

Fig. 7. Localization errors with k-NN and adaptive k-NN.



(a) Impacts of number. (b) Impacts of depth.

Fig. 8. Detection performance of different parameters.

goes to be flattened even the scale of nearest neighbors are relatively small. Nevertheless, decreasing k in the classic k-NN algorithm could achieve a high prediction accuracy, but will result in higher complexity of the model, and sometimes even the over-fitting phenomena. To balance the accuracy and expense of module's generalization, we empirically set the number of nearest neighbors to 80, and the corresponding grid localization model can achieve a mean accuracy of 91.3%.

After determining which grid square the target aerial vehicle is in, GPS-probe estimates the fine-grained target position in the corresponding square. We analyze the errors of fine-grained localization model with adaptive k-NN, and compare them with standard k-NN algorithm. The results show that adaptive k-NN performs 16.5% better than the k-NN method with 8 nearest neighbors on mean errors. Fig. 7(a) shows the error comparision of the fine-grained localization models. We find that our localization approach does better than standard k-NN method on noisy RSSI measurements as those obtained from the real-world data. Since our approach is more robust against outliers, this is to be expected as the 99%ile of k-NN based localization is relatively higher than our approach.

Localization Based on TSS. As in the RSSI-based localization, we empirically set $k = 65$ in k-NN for grid labeling, and this model can achieve a mean accuracy of 95.1%. Likewise, we estimate the fine-grained target position within the located square using adaptive k-NN and standard k-NN method. The experiments show that adaptive k-NN performs 11.7% better than the k-NN method with 8 nearest neighbors on mean errors. From Fig. 7(b) we find that adaptive k-NN performs better than k-NN algorithm even when k changes. In general, regardless of the constant number of neighbors, an outlier could always induce a non-negligible localization bias to k-NN based localization algorithms. Our adaptive k-NN can filter out the outliers when selecting neighbors, thereby achieving a high localization accuracy.

C. Performance of Spoofing Detection

To show how well GPS-Probe performs with the real-world flight data, we focus on evaluating the performance of spoofing detection in terms of the detection accuracy and the detection delay.

Detection Accuracy. We first assess the impact of two major parameters in Xgboost algorithm: the number and the depth of the decision trees, denoted as N_{tree} and D_{tree} respectively. Concerning the choice of N_{tree}, Fig. 8(a) demonstrates the accuracy gained with the increasing number of decision trees. It is observed that a large accuracy improvement eases until $N_{tree} = 160$. Further decreases in mean accuracy are small and much less pronounced with more decision trees. As for the choice of D_{tree}, Fig. 8(b) shows that a large accuracy improvement eases until $D_{tree} = 9$. These experiments show that we can achieve a high detection accuracy with a set of proper parameters of Xgboost.

Subsequently, we evaluate the performance of GPS-Probe with different position deviations caused by GPS spoofing attacks. In order to simulate the spoofed GPS positions, we add different localization deviations on a set of randomly selected flight data. Specifically, we add deviations from $1km$ to $10km$ to the original GPS coordinates, and analyze the detection performance of GPS-Probe under different spoofing situations. The experiment results show that the detection accuracy and precision is up to 64.5% and 70.4% respectively when the spoofed GPS position has a deviation of $1km$ to its real position. Fig. 9 shows that the increasing of spoofed position deviation makes it easier to detect attacks in GPS-Probe. All the detection accuracy, precision and recall can reach around 90% with a mean GPS-derived position spoofed bias of $10km$.

Detection Delay. We define the detection delay as the time spanning from the time the ATC message is received by the OpenSky data server to the time GPS-Probe outputs the detection result. The detection delay corresponds to the running time of GPS-Probe, which is induced by the computation of localization and spoofing detection. Once our localization and detection models are constructed, GPS-Probe can achieve an average detection delay close to $1.7s$ as shown in Fig. 11.

D. Comparison Results

We compare the detection accuracy and detection delay of GPS-Probe with Crowd-GPS-Sec in this section. Since Crowd-GPS-Sec cannot detect spoofing attacks with less than four ground sensors, we select the ATC data which are monitored by at least four ground sensors to conduct our comparison experiments. The settings of the comparison such as airspace density and ground sensor density remain consistent with [15]. The mean detection accuracy of the two algorithms is shown in Fig. 10. It shows that GPS-Probe apparently outperforms Crowd-GPS-Sec, and achieves the mean detection rate of 81.7%, which is better than Crowd-GPS-Sec by about 28.2%.

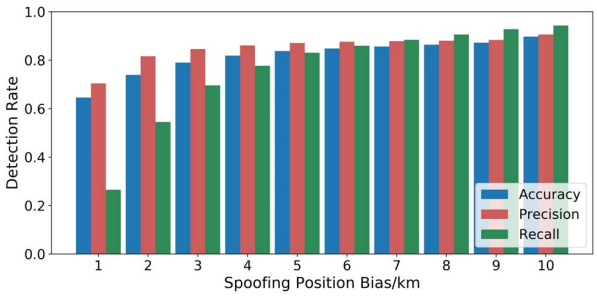


Fig. 9. Detection accuracy of different spoofing attack ranges.

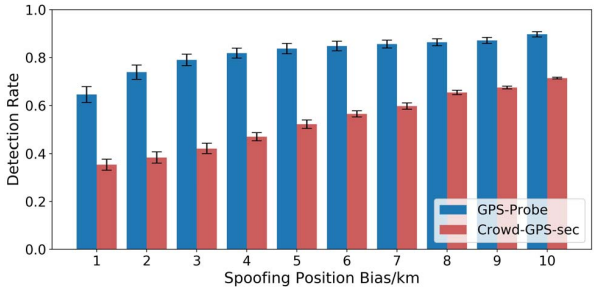


Fig. 10. Comparison of the detection accuracy.

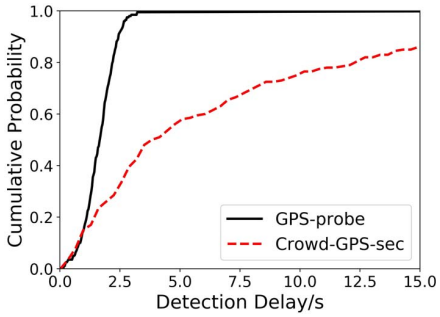


Fig. 11. Comparison of the detection delay.

More specifically, GPS-Probe can achieve a detection accuracy of 64.5% with $1km$ spoofing position bias, while Crowd-GPS-Sec only achieves 35.4%. Even with $10km$ attacked position bias, the detection accuracy of GPS-Probe achieves 90%, which outperforms Crowd-GPS-Sec by 18.3%. The results are not surprising since our detection model based on machine learning has the ability to "learn" different attack modes, while Crowd-GPS-Sec can only detect spoofing attacks with a constant threshold. Fig. 11 further shows that the detection delay of Crowd-GPS-Sec is much longer than GPS-Probe. The average delay of Crowd-GPS-Sec is $5.3s$ due to long TDoA-based multilateration time. In contrast, GPS-Probe can get the estimated position within $1s$ and achieve a mean spoofing detection delay of $1.7s$.

V. Related Works

Detection of GPS spoofing attacks has received considerable attentions both in the industry and academia. Existing detection methods can be coarsely categorized into three class. The first class leverages the characteristics of RF signals, such as signal strength, signal peaks, and signal phase. Akos [14] suggests to monitor the incoming GPS signal strength and the state of the automatic gain control. Another technique called SPREE relies on auxiliary peak tracking [25] to detect suspicious peaks from signals with weaker acquisition correlation peaks. Psiaki et al. [26] propose a detection scheme which uses an additional reference receiver to correlate the received signal with authentic signals assuming the inclusion of the encrypted military signal. A spoofed signal does not correlate with the reference node's received signal and the attack can be detected.

The second class of detection approaches relies on multiple receiving antennas or multiple receivers. Montgomery et al. [11] use a dual antenna receiver setup to determine the angle of arrival of incoming signals, and Psiaki et al. [27] extended this approach to include differential carrier phase measurements. Multiple co-located GPS receivers are used in [3], [28] to detect attacks by comparing the GPS coordinates and times of these receivers. The use of receiving antenna arrays shows that signal diversity of different antennas is an effective indicator to detect spoofing attacks without preknowledge of the target's location [29]. Although these detection approaches do not require upgrades of the existing GPS infrastructure, they need more sophisticated GPS receivers which would significantly increase the costs, complexity and power requirements.

Since the massive upgrades of GPS infrastructure are unlikely to happen in the near future, Jansen et al [15] proposed a novel method called Crowd-GPS-Sec to detect spoofing attacks. Crowd-GPS-Sec leverages the ATC messages of aerial vehicles which are periodically broadcasted for ATC purpose. It locates a target aerial vehicle by an independent infrastructure on the ground which analyzes the contents and the times of arrival of these GPS broadcasts. Then it compares the location results with the GPS-derived position to identify whether the target is attacked. However, the performance of Crowd-GPS-Sec depends heavily on time synchronization of the ground infrastructure as well as on the location accuracy of GPS devices.

VI. Conclusion

In this paper we have presented GPS-Probe, a synchronization-free GPS spoofing detection scheme with crowdsourced ATC data. GPS-Probe is the first work to take advantage of RSSI and TSS features in crowdsourced ATC data, and leverage machine learning techniques for GPS spoofing detection. It neither requires any updates of the GPS infrastructure nor updates of the GPS receivers. More importantly, it releases the requirement on the time synchronization of the ground sensors distributed around the world. Our evaluation results with real-world ATC data from the OpenSky Network validate its effectiveness and efficiency.

REFERENCES

[1] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, *Global positioning system: theory and practice*. Springer, 2012.

[2] C. Wang, H. Lin, and H. Jiang, "CANS: Towards congestion-adaptive and small stretch emergency navigation with wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 5, pp. 1077–1089, 2016.

[3] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Čapkun, "On the requirements for successful GPS spoofing attacks," in *Proceedings of ACM CCS*, 2011, pp. 75–85.

[4] D. Moser, P. Leu, V. Lenders, A. Ranganathan, F. Ricciato, and S. Capkun, "Investigation of multi-device location spoofing attacks on air traffic control and possible countermeasures," in *Proceedings of ACM MobiCom*, 2016, pp. 375–386.

[5] T. E. Humphreys, B. M. Ledvina, M. L. Psiaki, B. W. O'Hanlon, and P. M. Kintner, "Assessing the spoofing threat: Development of a portable GPS civilian spoofer," in *Proceedings on Radionavigation Laboratory Conference*, 2008.

[6] A. J. Kerns, D. P. Shepard, J. A. Bhatti, and T. E. Humphreys, "Unmanned aircraft capture and control via GPS spoofing," *Journal of Field Robotics*, vol. 31, no. 4, pp. 617–636, 2014.

[7] M. L. Psiaki, T. E. Humphreys, and B. Stauffer, "Attackers can spoof navigation signals without our knowledge. here's how to fight back GPS lies," *IEEE Spectrum*, vol. 53, no. 8, pp. 26–53, 2016.

[8] K. D. Wesson, J. N. Gross, T. E. Humphreys, and B. L. Evans, "GNSS signal authentication via power and distortion monitoring," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 2, pp. 739–754, 2018.

[9] L. Heng, D. B. Work, and G. X. Gao, "GPS signal authentication from cooperative peers," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 1794–1805, 2015.

[10] K. Wesson, M. Rothlisberger, and T. Humphreys, "Practical cryptographic civil GPS signal authentication," *Navigation: Journal of The Institute of Navigation*, vol. 59, no. 3, pp. 177–193, 2012.

[11] P. Y. Montgomery, "Receiver-autonomous spoofing detection: Experimental results of a multi-antenna receiver defense against a portable civil GPS spoofer," in *Proceedings of Radionavigation Laboratory Conference*, 2011.

[12] J. Nielsen, A. Broumandan, and G. Lachapelle, "GNSS spoofing detection for single antenna handheld receivers," *Navigation*, vol. 58, no. 4, pp. 335–344, 2011.

[13] M. L. Psiaki, S. P. Powell, and B. W. Ohanlon, "GNSS spoofing detection using high-frequency antenna motion and carrier-phase data," in *Proceedings of the ION GNSS Meeting*, 2013, pp. 2949–2991.

[14] D. M. Akos, "Who's afraid of the spoofer? GPS/GNSS spoofing detection via automatic gain control (AGC)," *Navigation: Journal of the Institute of Navigation*, vol. 59, no. 4, pp. 281–290, 2012.

[15] K. Jansen, M. Schäfer, D. Moser, V. Lenders, C. Pöpper, and J. Schmitt, "Crowd-GPS-Sec: Leveraging crowdsourcing to detect and localize GPS spoofing attacks," in *Proceedings of IEEE S&P*, 2018.

[16] B. Xu, G. Sun, R. Yu, and Z. Yang, "High-accuracy TDOA-based localization without time synchronization," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 8, pp. 1567–1576, 2013.

[17] M. Schfer, M. Strohmeier, V. Lenders, I. Martinovic, and M. Wilhelm, "Bringing up OpenSky: A large-scale ADS-B sensor network for research," in *Proceedings of ACM/IEEE IPSN*, 2014, pp. 83–94.

[18] M. Nolan, *Fundamentals of Air Traffic Control*. Cengage learning, 2010.

[19] R. Trüb, D. Moser, M. Schäfer, R. Pinheiro, and V. Lenders, "Monitoring meteorological parameters with crowdsourced air traffic control data," in *Proceedings of ACM/IEEE IPSN*, 2018, pp. 25–36.

[20] Y. Zheng, Y. Liu, and Z. Zhou, "From RSSI to CSI: Indoor localization via channel response," *ACM Computing Surveys*, vol. 46, no. 2, pp. 1–32, 2013.

[21] X. Liu, J. Cao, S. Tang, J. Wen, and P. Guo, "Contactless respiration monitoring via off-the-shelf WiFi devices," *IEEE Transactions on Mobile Computing*, vol. 15, no. 10, pp. 2466–2479, 2016.

[22] M. Strohmeier, V. Lenders, and I. Martinovic, "A localization approach for crowdsourced air traffic communication networks," *arXiv preprint arXiv:1610.06754*, 2016.

[23] J.-A. Hernández and I. W. Phillips, "Weibull mixture model to characterise end-to-end internet delay at coarse time-scales," *IEE Proceedings-Communications*, vol. 153, no. 2, pp. 295–304, 2006.

[24] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of ACM SIGKDD*, 2016, pp. 785–794.

[25] A. Ranganathan, H. Ólafsdóttir, and S. Capkun, "Spree: A spoofing resistant GPS receiver," in *Proceedings of MobiCom*, 2016, pp. 348–360.

[26] M. L. Psiaki, B. W. O'Hanlon, J. A. Bhatti, D. P. Shepard, and T. E. Humphreys, "GPS spoofing detection via dual-receiver correlation of military signals," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, no. 4, pp. 2250–2267, 2013.

[27] M. L. Psiaki, B. W. O'hanlon, S. P. Powell, J. A. Bhatti, K. D. Wesson, and T. E. Humphreys, "GNSS spoofing detection using two-antenna differential carrier phase," in *Proceedings of Radionavigation Laboratory Conference*, 2014.

[28] K. Jansen, N. O. Tippenhauer, and C. Pöpper, "Multi-receiver GPS spoofing detection: error models and realization," in *Proceedings of ACM ACSAC*, 2016, pp. 237–250.

[29] J. Magiera and R. Katulski, "Detection and mitigation of GPS spoofing based on antenna array processing," *Journal of Applied Research and Technology*, vol. 13, no. 1, pp. 45–57, 2015.