This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3601022

Machine Learning-Aided Localization-Based Attack Detection in Movable Antenna Systems

Tiep M. Hoang and Alireza Vahid

***Abstract*—Physical-layer (PHY) attacks increasingly pose a threat to location-based services. To secure the localization mechanism against PHY attacks, we propose a novel framework based on localization and user-and-attacker detection, with the help of unsupervised machine learning (ML) algorithms and multiple signal classification (MUSIC) spectra. Our proposed framework consists of two stages: i) uplink localization and detection; and ii) downlink secure transmission. Noticeably, in the proposed framework, a reciprocal relationship between the localization mechanism and the user/attacker detection is developed, where the localization supports the detection and vice versa. This reciprocal relationship allows wireless systems to detect localization attacks and further localize the attacker. Through simulation, we show the efficacy of combining localization and detection in the uplink. We then demonstrate the benefit of employing both localization and movable-antenna arrays for secure downlink transmission.**

***Index terms*—**Physical layer security, anomaly detection, localization, machine learning, multi-user detection, MUSIC spectrum.

I. Introduction

Location-based services (LBSs) like location-based advertisements and tracking systems have been more and more popular and ubiquitous in many aspects of everyday life. The systems that provide LBSs include global navigation satellite systems, IEEE 802.11 (i.e., Wi-Fi) systems and cellular systems. In general, wireless signals can be used by different wireless networks for performing localization and providing LBSs [1], [2]. As expected, wireless localization systems will also suffer from multi-user interference and physical-layer (PHY) attacks. Thus, it is worth designing a robust localization system that can alleviate the negative impact of interference, as well as PHY attacks, in order to enhance localization performance. In doing so, an important task is to detect multiple users and PHY attacks at the same time, as well as being able to differ PHY attacks from interference.

When it comes to wireless intrusion detection, some recent advanced methods rely on machine learning algorithms [3], [4]. While ML can be categorized into many categories, unsupervised learning seems to be suitable for the case of not having prior-knowledge of adversaries/attackers. However, so far, there has been no clear usage of ML in detecting PHY attacks in localization systems. Motivated by this, we will consider integrating unsupervised ML algorithms into our proposed localization system in this paper. It should also be noted that the usage of ML normally comes with the creation of suitable datasets for training and testing. In the case of localization, PHY characteristics like time-of-arrival (TOA), angle-of-arrival (AOA), and received signal strength (RSS) can be used as fingerprints for the classification purposes, as can be exemplified in [5]. Thus, it is also an open question of how unsupervised ML can support a localization system against interference and PHY attacks.

T. M. Hoang and A. Vahid are with the Department of Electrical and Microelectronic Engineering, Rochester Institute of Technology, NY 14623, USA (emails: tmheme@rit.edu, arveme@rit.edu).

This material is based upon work supported by the National Science Foundation under Award No. CNS-2343964. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

A. Main Challenges and Contributions

Although both secure and robust localization have received growing attention, most existing works either focus on improving positioning accuracy under normal conditions or address intrusion detection separately from localization. In practice, interference and PHY attacks often appear together, and their effects are difficult to distinguish. This creates a dual challenge for both system security and localization accuracy. Localization algorithms that overlook adversarial behavior may produce misleading results, reducing the reliability of LBSs.

To address this challenge, it is important to develop a unified framework that can detect anomalies while maintaining accurate localization. This is especially relevant in multi-user uplink settings, where the presence of multiple users transmitting at once complicates the task of identifying and isolating a potential attack from legitimate signals. The combined use of machine learning and signal processing methods (such as ML-aided anomaly detection and MUSIC-based distance estimation) offers a promising solution.

Motivated by these challenges, our work introduces a reciprocal integration of localization and detection mechanisms that support and enhance each other, resulting in a more resilient wireless system. The contributions of this paper are summarized as follows:

- We consider the localization and security aspects of a wireless system, where the localization process undergoes PHY attacks in the uplink. To protect the system against PHY attacks, we propose a joint localization-detection method, where localization and detection support each other in a reciprocal manner. Note that the concept of detection encompasses both multi-user detection (MUD) and attacker detection.
- As for user localization, we use the trilateration technique that relies on estimated distances from legitimate users to access points (APs). To consolidate the localization

performance in the case of multi-user uplink, we propose an iterative algorithm for the distance estimation process (DEP), which can respectively extract the distances for each user by combining an MUD technique and multiple MUSIC spectra inside the DEP.
- As for attacker detection, we employ anomaly detection algorithms for classifying normal received signals and abnormal ones. Furthermore, the detection of PHY attacks not only improves the reliability of the localization mechanism, but also allows for localizing the attacker and removing its negative impact. We consider two different data structures and demonstrate the significant difference in overall performance when applying anomaly detection algorithms to those datasets, highlighting the importance of feature selection and structure in our learning-aided localization-detection method.
- Finally, we demonstrate the benefit of our secured localization mechanism by evaluating the performance of position-aware transmission in the downlink, especially with the use of movable-antenna arrays at the APs.[1]

B. Related works and comparisons

In the context of PHY authentication, most previous works adopt non-learning approaches based on hypothesis testing. For example, the authors of [6] propose a PHY authentication scheme for a UAV-based mmWave system by leveraging image processing techniques to model angular-domain channels. Authentication performance is evaluated through hypothesis testing, with an approximation method applied to determine the detection threshold. Considering the 5G NR specifications, [7] investigates attacks on the physical broadcast channel block and proposes a hypothesis test for detection. A hypothesis testing problem is also formulated in [8], where the authors focus on detecting attacks in RIS-aided systems by extracting features from both the direct and cascaded channels. Additionally, auxiliary information from second-order statistics is leveraged to enhance the authentication process. In [9], a binary hypothesis test is formulated to authenticate groups of backscatter devices by analyzing the sum of differences across multiple channel impulse responses. More recently, [10] proposes a PHY authentication method for mmWave systems based on propagation features and hardware impairments. It uses binary hypothesis testing with weighted decision fusion to detect spoofing. In general, hypothesis testing-based methods rely on predefined decision rules and require prior knowledge of both legitimate and attacker signal distributions. In contrast, our proposed framework employs unsupervised learning methods that can be trained solely on legitimate data, thus enabling the detection of unseen attacks without fixed decision rules.

In the context of localization, it remains an important topic because of its wide practical applications. Recent studies include [5], [11]–[16]. In particular, [11] first presents the use of a pair of RISs for channel estimation, and then harnesses the estimated channel parameters for user localization. In spite of the high complexity, the framework in [11] is only considered for the single-user uplink scenario, making it difficult to be applied to multi-user uplink. In [5], the authors consider the information of angles, delays and powers as the positioning fingerprints of users and train a convolutional neural network on the fingerprint dataset to localize users. [12] considers an integrated sensing and communication (ISAC) system, where mmWave communication signals are also leveraged for multi-user location sensing. Also in [12], under the assumption that there is no direct path between transceivers, the authors introduce RISs as intermediate relays for creating new paths and AOAs, which will then be estimated and employed for user localization. The authors of [13] first consider the deployment of mmWave 5G NR systems in industrial environments and then propose a joint multi-user positioning and clock synchronization based on TOA and AOA measurements. [14] proposes a joint localization and communication framework, where time slots are respectively allocated for localization and communication purposes. Similar to [11] and [12], the framework in [14] also relies on RISs to support transmission under the assumption that the line-of-sight (LoS) path between transceivers is obstructed. In [16], the authors present a method of tracking targets based on simultaneously localizing and detecting them. Considering real-time positioning services in practical environments, [15] develops a testbed to validate the feasibility of AoA-based localization in ultra dense networks.

All the aforementioned works [5], [11]–[16] only focus on improving localization performance in benign environments where no adversary exists. Moreover, given that wireless propagation is susceptible to PHY attacks, it is also crucial to consider the localization performance degradation under PHY attacks and the countermeasures against them. In this vein, only a few recent works address the need for robust localization mechanisms against PHY attacks [17]–[19]. To be more specific, the authors of [17] develops an anomaly-detection-based framework for enhancing the reliability of LoS-AoA estimation and eliminating low-quality data during the localization process. Meanwhile, the authors of [18] focus on dealing with cooperative localization attacks. The authors of [19] consider a two-way TOA positioning process and propose a security solution against the attacks that corrupt distance measurements at anchor nodes. However, the security solution [19] is not for the case of multiple users; and yet, the attacker detection method is based on classical hypothesis testing rather than ML models. Some recent studies have partially explored the impact of location on security and authentication performance [20]–[22]. For example, [20] presents an authentication framework to protect an unmanned aerial vehicle communication system by distinguishing jamming attacks from legitimate signals. In [21], although localization attacks are not the primary focus, co-located attacks are partially addressed through a physical-layer authentication framework that leverages intrinsic hardware characteristics and analyzes beam pattern deviations. In contrast to these works, [22] proposes an attack rather than a security method, using reinforcement learning to track the locations of mobile devices based on RSS indicator data. Noticeably, none of the works

[1]The use of movable-antenna arrays for localization and detection in the uplink may be complicated because the rotation of movable-antenna arrays will make it hard to estimate AoAs, which are inherently random in multi-path environments. Thus, we leave this for future works and instead consider movable-antenna arrays in the downlink.

[17]–[22] apply anomaly detection algorithms for attacker detection.

Different from the works in [5], [11]–[16] that ignore security threats in user positioning, we consider the security threats caused by adversaries and design a robust localization mechanism against PHY attacks. Additionally, compared to the related works [17]–[19] that partially touch upon reliable localization, our work considers the use of anomaly detection algorithms in detecting PHY-attack-related anomalies and recovering the reliability of the localization mechanism. We note that [17] considers the identification of anomalies in the AoA-based localization mechanism, but does not focus on protecting positioning systems against PHY attacks. In [18], a method of detecting PHY localization attacks is proposed, but ML is not applied. By contrast, we consider the application of ML algorithms in detecting PHY attacks. Finally, while the joint topic of localization and detection has been witnessed in previous works [16]–[19], these works do not build a reciprocal relationship between localization and detection. Our work, however, establishes a mutually supportive relationship between localization and detection mechanisms. In short, while previous works often address localization and attack detection as separate tasks, our approach integrates them into a unified framework where each function enhances the other. This integration not only improves the robustness and security of wireless localization systems but also advances the role of ML in enabling this reciprocal framework.

C. Organization and Notations

Organization: The rest of the paper is organized as follows. Section II presents the proposed system model, the problem statement and the proposed framework. Section III presents how to process the received signals at access points for uplink localization and detection. In Section IV, we address the detection of PHY attacks based on ML and the positioning of the attacker based on statistical knowledge of users' positions. In Section V, we first present numerical results related to uplink localization and detection, and then demonstrate the efficiency of our proposed method through evaluating the position-aware payload transmission in the downlink. Finally, Section VI concludes the paper.

Notations: $\mathbb{R}^{m\times n}$ denotes the real field that includes all real-valued matrices of size $m\times n$; $\mathbb{C}^{m\times n}$ denotes the complex field that includes all complex-valued matrices of size $m\times n$; Bold lowercase letters and bold uppercase letters denote vectors and matrices, respectively; $\mathbf{I}_n$ denotes the identity matrix of size $n\times n$; The superscripts $(\cdot)^\top$, $(\cdot)^*$, and $(\cdot)^\dagger$ represent the transpose, conjugate, and Hermitian operators, respectively; $\mathbf{z}\sim\mathcal{CN}(\mathbf{m},\mathbf{\Sigma})$ is a complex Gaussian random vector with mean $\mathbf{m}$ and covariance matrix $\mathbf{\Sigma}$.

II. Modeling and Problem Statement

We consider a wireless system that consists of three access points (APs) cooperating with each other to perform localization and detection in the uplink. When U legitimate users (namely, B_u) transmit their signals to the APs, the adversary (namely, T) also sends jamming (or spoofing) signals to interfere with (or to deceive) the APs, whereby deteriorating the reception and localization processes. The positions of the APs and the position of the attacker are assumed to be fixed. By contrast, the users move around and their positions are spatially distributed around some landmarks. We assume that there are M receive antennas at each AP, while there is a single antenna at each user and the adversary. Due to the presence of the adversary, we have two hypotheses: i) under $(\mathcal{H}_0)$, there is no attack; ii) under $(\mathcal{H}_1)$, there is an attack.

Regarding signal modeling, we denote K_{sub} as the number of subcarriers. The set of frequencies is $\mathcal{F}=\{f_c, f_c+\Delta_F,\ldots,f_c+(K_{\text{sub}}-1)\Delta_F\}$, where the frequency spacing Δ_F between two adjacent subcarriers is a constant. Due to scattering, there are L_{paths} paths between a legitimate transmitter and a receiver. A signal that originates from a user and follows the ℓ-th path will be associated with the TOA $\tau_{\ell i}$ and the AOA $\theta_{\ell i}$. Denote $\{\tau_1,\ldots,\tau_{L_{\text{paths}}}\}$ as the set of ToAs, and $\{\theta_1,\ldots,\theta_{L_{\text{paths}}}\}$ as the set of AoAs. We will delineate that $\ell=1$ indicates the LoS path and $\ell>1$ indicates NLoS paths.

A. Signal Modeling for Single-User Uplink

Before presenting the case of multi-user uplink, we first present the case of a single user to clarify concepts and terminologies in this sub-section. In the case that only the u-th user transmits its signals while other nodes (including the adversary) do not transmit anything, the received signals, at the m-th receive antenna ($m\in\{1,\ldots,M\}$) of the i-th AP and over all K_{sub} subcarriers, are arranged in the vector $\mathbf{r}_{u\to m,i|\mathcal{H}_0}(t)\in\mathbb{C}^{K_{\text{sub}}\times 1}$. The explicit expression of $\mathbf{r}_{u\to m,i|\mathcal{H}_0}(t)$ can be given by (1), as shown at the bottom of the next page. In (1), $s(t)\in\mathbb{C}$ is the transmitted signal, P_{Tx} is the transmit power, $\mathfrak{L}_i^{(\ell)}$ is the path loss that corresponds to the ℓ-th path, and $\mathbf{n}_m(t)\in\mathbb{C}^{K_{\text{sub}}\times 1}$ is the additive white Gaussian noise (AWGN) vector. Note that $s(t)$ is normalized so that with $\mathbb{E}\left\{|s(t)|^2\right\}=1$, while the average power of each element of $\mathbf{n}_m(t)$ is N_0.

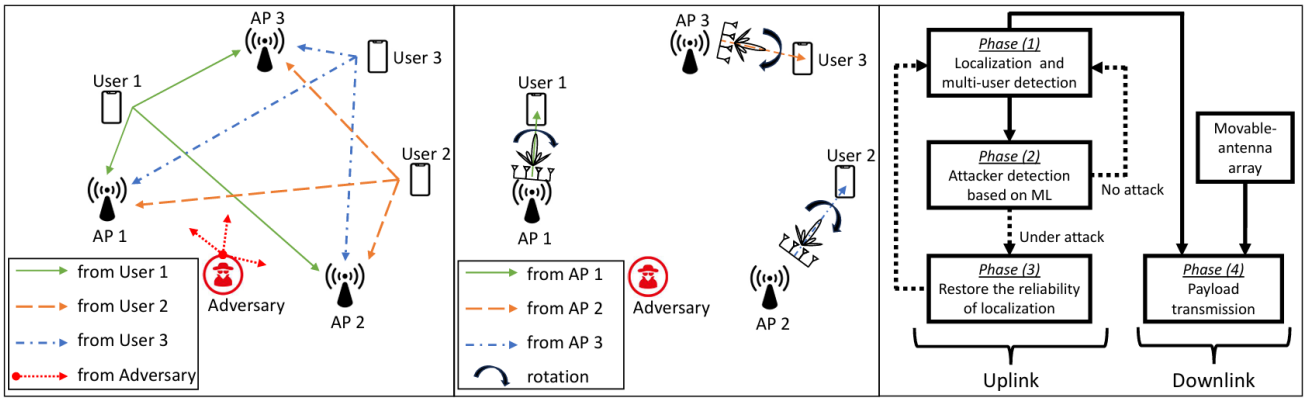
Still concerning the hypothesis $(\mathcal{H}_0)$, the received signals at the i-th AP (over all M receive antennas and K_{sub} subcarriers) can be arranged in the following vector:

$$\mathbf{r}_{u\to i|\mathcal{H}_0}(t)=\left[\mathbf{r}_{u\to 1,i|\mathcal{H}_0}^\top(t),\ldots,\mathbf{r}_{u\to M,i|\mathcal{H}_0}^\top(t)\right]^\top\in\mathbb{C}^{MK_{\text{sub}}\times 1}. \tag{2}$$

Define $\mathbf{v}(\tau_{\ell i})\triangleq\left[1,e^{-j2\pi\Delta_F\tau_{\ell i}},\ldots,e^{-j2\pi\Delta_F(K_{\text{sub}}-1)\tau_{\ell i}}\right]^\top$, $\psi_m(\theta_{\ell i})\triangleq e^{-j2\pi\frac{f_c d_{\text{spacing}}}{c}(m-1)\sin(\theta_{\ell i})}$, $s_{(u,\ell)}(t)\triangleq\sqrt{\frac{P_{\text{Tx}}}{\mathfrak{L}_i^{(\ell)}N_0}}s(t)$, and $\mathbf{n}(t)=\left[\mathbf{n}_1^\top(t),\ldots,\mathbf{n}_M^\top(t)\right]^\top$. We can re-write $\mathbf{r}_{u\to i|\mathcal{H}_0}(t)$ in (2) as follows:

$$\begin{aligned}\mathbf{r}_{u\to i|\mathcal{H}_0}(t)&=\sum_{\ell=1}^{L_{\text{paths}}}\begin{bmatrix}\psi_1(\theta_{\ell i})\mathbf{v}(\tau_{\ell i})\\ \vdots\\ \psi_M(\theta_{\ell i})\mathbf{v}(\tau_{\ell i})\end{bmatrix}s_{(u,\ell)}(t)e^{-j2\pi f_c\tau_{\ell i}}+\mathbf{n}(t)\\&=\sum_{\ell=1}^{L_{\text{paths}}}\mathbf{a}(\tau_{\ell i},\theta_{\ell i})s_{(u,\ell)}(t)e^{-j2\pi f_c\tau_{\ell i}}+\mathbf{n}(t),\end{aligned} \tag{3}$$

where $\mathbf{a}(\tau,\theta)\triangleq\left[\psi_1(\theta)\mathbf{v}^\top(\tau),\ldots,\psi_M(\theta)\mathbf{v}^\top(\tau)\right]^\top$. Each element in $\mathbf{a}(\tau,\theta)$ is a function of τ and θ. Note that the constant



(a) Uplink (b) Downlink (c) Proposed framework

Fig. 1: The proposed system model and framework are depicted. The sub-figure (a) depicts the uplink stage in the case that there are multiple users and an adversary. The sub-figure (b) depicts the usage of movable-antenna arrays in the downlink stage, when the APs are aware of the positions of nodes. The sub-figure (c) depicts the proposed framework that includes related processes such as localization, user and attacker detection.

vector $\mathbf{a}(\tau_{\ell i},\theta_{\ell i})$ in (3) is $\mathbf{a}(\tau_{\ell i},\theta_{\ell i}) = \mathbf{a}(\tau,\theta)\big|_{\tau=\tau_{\ell i},\theta=\theta_{\ell i}}$. The expression in (3) is a mathematical model of how a wireless signal is received after propagation. It captures how the signal is affected by distances (modeled through time delays), directions (modeled through AoAs), and noise as it travels from a transmitter (i.e., B_u) to a receiver (i.e., the i-th AP) along multiple paths.

Recall that the expression of the received signal in (3) holds true for the case of no attack (i.e., under $\mathcal{H}_0$). By contrast, under the assumption $(\mathcal{H}_1)$ that there is an attack from the adversary, the received signal can be given by

$$\mathbf{r}_{u\to i|\mathcal{H}_1}(t) = \sum_{\ell=1}^{L_{\text{paths}}} \mathbf{a}(\tau_{\ell i},\theta_{\ell i}) s_{(u,\ell)}(t) e^{-j2\pi f_c \tau_{\ell i}} + \underbrace{\sum_{\ell'=1}^{L_{\text{jam}}} \mathbf{a}(\tau_{\ell' i},\theta_{\ell' i}) \widetilde{s}_{\ell'}(t) e^{-j2\pi f_c \tau_{\ell' i}}}_{\text{related to attacker}} + \mathbf{n}(t), \tag{4}$$

where L_{jam} is the number of paths from the jammer to the receiver, and $\widetilde{s}_{\ell'}(t)$ is defined as $\widetilde{s}_{\ell'}(t) \triangleq \sqrt{\frac{P_{\text{jam}}}{\mathfrak{L}_i^{(\ell')} N_0}}\widetilde{s}(t)$. Herein, $\widetilde{s}(t)$ is the signal transmitted by the attacker, P_{jam} is the transmit power of the attacker, and $\mathfrak{L}_i^{(\ell')}$ is the path loss corresponding to the ℓ'-th path, $\ell' \in \{1,\ldots,L_{\text{jam}}\}$.

To further simplify the expressions, we first define the following vectors and matrices:

$$\underline{\mathbf{a}}_\ell \triangleq \mathbf{a}(\tau_{\ell i},\theta_{\ell i}) e^{-j2\pi f_c \tau_{\ell i}}, \text{ for } \ell \in \{1,\ldots,L_{\text{paths}}\},$$

$$\widetilde{\underline{\mathbf{a}}}_{\ell'} \triangleq \mathbf{a}(\tau_{\ell' i},\theta_{\ell' i}) e^{-j2\pi f_c \tau_{\ell' i}}, \text{ for } \ell' \in \{1,\ldots,L_{\text{jam}}\},$$

$$\mathbf{H}_{u\to i} \triangleq \left[\underline{\mathbf{a}}_1,\ldots,\underline{\mathbf{a}}_{L_{\text{paths}}}\right],$$

$$\widetilde{\mathbf{H}}_i \triangleq \left[\widetilde{\underline{\mathbf{a}}}_1,\ldots,\widetilde{\underline{\mathbf{a}}}_{L_{\text{jam}}}\right],$$

$$\mathbf{s}_u(t) \triangleq \left[s_{(u,1)}(t),\ldots,s_{(u,L_{\text{paths}})}(t)\right]^\top,$$

$$\widetilde{\mathbf{s}}(t) \triangleq \left[\widetilde{s}_1(t),\ldots,\widetilde{s}_{L_{\text{jam}}}(t)\right]^\top.$$

Then, (3) and (4) can be shorten into the following:

$$\mathbf{r}_{u\to i|\mathcal{H}_0}(t) = \mathbf{H}_{u\to i}\mathbf{s}_u(t) + \mathbf{n}(t), \tag{5}$$

$$\mathbf{r}_{u\to i|\mathcal{H}_1}(t) = \mathbf{H}_{u\to i}\mathbf{s}_u(t) + \widetilde{\mathbf{H}}_i\widetilde{\mathbf{s}}(t) + \mathbf{n}(t). \tag{6}$$

B. Signal Modeling for Multi-User Uplink

The expressions in (3) and (4) only hold true for the case of a single user. However, when $U \geq 2$ users are transmitting simultaneously, the received signals have to be revised. Under $\mathcal{H}_0$, the received signal at the i-th AP, can be given by

$$\begin{aligned}\mathbf{r}_{i\mathcal{H}_0}(t) &= \sum_{u=1}^{U}\left(\sum_{\ell=1}^{L_{\text{paths}}}\underline{\mathbf{a}}_\ell s_{(u,\ell)}(t)\right) + \mathbf{n}(t)\\ &= \mathbf{H}_{1\to i}\mathbf{s}_1(t) + \ldots + \mathbf{H}_{U\to i}\mathbf{s}_U(t) + \mathbf{n}(t)\\ &= \mathbf{H}_i\mathbf{s}(t) + \mathbf{n}(t),\end{aligned} \tag{7}$$

where $\mathbf{H}_i, \mathbf{s}_u(t), \mathbf{s}(t)$ are defined as

$$\mathbf{H}_i \triangleq \left[\mathbf{H}_{1\to i},\ldots,\mathbf{H}_{U\to i}\right],$$

$$\mathbf{s}(t) \triangleq \left[\mathbf{s}_1^\top(t),\ldots,\mathbf{s}_U^\top(t)\right]^\top.$$

$$\mathbf{r}_{u\to m,i|\mathcal{H}_0}(t) = \sum_{\ell=1}^{L_{\text{paths}}} \underbrace{\begin{bmatrix}1\\ e^{-j2\pi\Delta_F\tau_{\ell i}}\\ \vdots\\ e^{-j2\pi\Delta_F(K_{\text{sub}}-1)\tau_{\ell i}}\end{bmatrix}}_{\text{for } K_{\text{sub}} \text{ subcarriers}} \underbrace{e^{-j2\pi\frac{f_c d_{\text{spacing}}}{c}(m-1)\sin(\theta_{\ell i})}}_{\text{for the } m-\text{th antenna element}} \underbrace{\sqrt{\frac{P_{\text{Tx}}}{\mathfrak{L}_i^{(\ell)}N_0}}s(t)e^{-j2\pi f_c\tau_{\ell i}}}_{\text{for the } \ell-\text{th path}} + \underbrace{\mathbf{n}_m(t)}_{\text{noise over } K_{\text{sub}} \text{ subcarriers}}. \tag{1}$$

Similarly, under $\mathcal{H}_1$, the received signal at the i-th AP can be given by

$$\mathbf{r}_{i\mathcal{H}_1}(t) = \sum_{u=1}^{U}\left(\sum_{\ell=1}^{L_{\text{paths}}}\underline{\mathbf{a}}_\ell s_{(u,\ell)}(t)\right) + \underbrace{\sum_{\ell'=1}^{L_{\text{jam}}}\widetilde{\underline{\mathbf{a}}}_{\ell'}\widetilde{s}_{\ell'}(t)}_{\text{related to attacker}} + \mathbf{n}(t)$$
$$= \mathbf{H}_i\mathbf{s}(t) + \widetilde{\mathbf{H}}_i\widetilde{\mathbf{s}}(t) + \mathbf{n}(t). \tag{8}$$

Note that (5) and (6) are the special cases of (7) and (8), when the number of users is set to 1. Moreover, from (7) and (8), we can generally express the received signal vector at the i-th AP as follows:

$$\mathbf{r}_i(t) = \begin{cases}\mathbf{r}_{i\mathcal{H}_0}(t), & \text{under } (\mathcal{H}_0);\\ \mathbf{r}_{i\mathcal{H}_1}(t), & \text{under } (\mathcal{H}_1).\end{cases} \tag{9}$$

C. Problem Statement and Proposed Framework

Since the process of localizing users and the process of user and attacker detection occur in the uplink, a question may be raised: Can these processes be combined into a single framework, where both localization and user/attacker detection support each other? Especially, in the presence of multiple users and an adversary, the attack from the adversary can be mistaken for multi-user interference, making it difficult to detect the adversary. Thus, this paper aims to clarify this question and propose a framework for building the reciprocal relationship between localization and detection against PHY attacks with the aid of ML.

The proposed framework, which is illustrated in the third sub-figure of Figure 1, encompasses four main phases. Phase (1) performs a joint localization and multi-user detection; Phase (2) involves the use of ML to detect potential PHY localization attacks; Phase (3) addresses the restoration of reliable localization in response to detected attacks; and Phase (4) evaluates downlink performance based on the results from the previous phases. As previously mentioned, the interplay between localization and detection is central to the proposed framework, because they reinforce each other. To be more specific, as for localization, we first estimate the distances between the user and the access points based on the time-of-arrivals (TOAs) of signals and then use the trilateration technique to estimate the positions of users. As for detection, we apply different anomaly detection algorithms to learn the normal data and identify outliers (i.e., anomalies), whereby detecting if the received signals are associated with an attack or not. Since the detection returns one of the two results, i.e., no attack ($\mathcal{H}_0$) or under attack ($\mathcal{H}_1$), we can decide whether or not the localization is deteriorated by attacks. Thus, in the case of an attack, we can localize the position of the attacker.

At a higher level, the proposed framework demonstrates how integrating anomaly detection with PHY signal processing can support secure and robust localization in wireless systems. For instance, in location-aware downlink transmission, accurate localization can ensure efficient signal delivery. This integration is adaptable to various networks, where accurate positioning and resilience against interference or malicious attacks are essential.

III. Phase (1): Localization and Multi-user Detection

In this section, we present Phase (1) of the proposed framework, in which localization and multi-user detection are jointly performed.

A. Localization

1) Distance Estimation based on MUSIC Spectrum: Denote T_{samples} as the number of samples received at an AP. With respect to (w.r.t.) the i-th AP, the sample covariance can be calculated as follows:

$$\mathbf{C}_i = \frac{1}{T_{\text{samples}}}\sum_{t=1}^{T_{\text{samples}}}\mathbf{r}_i(t)\mathbf{r}_i^\dagger(t). \tag{10}$$

We can use the singular value decomposition (SVD) to decompose $\mathbf{C}_i$ to find a unitary matrix $\mathbf{U}_i$, and then extract the noise subspace $\mathbf{U}_i^{\text{noise}}$ from the last $(MK_{\text{sub}}-1)$ columns of $\mathbf{U}_i$. Consequently, at the i-th AP, the spectrum of MUSIC can be formulated as a function of the time delay τ and the angle θ (see [23], [24]):

$$S_i(\tau,\theta) = \frac{1}{\left|\mathbf{a}(\tau,\theta)\mathbf{U}_i^{\text{noise}}\right|^2}. \tag{11}$$

Figure 2 illustrates the use of (11) in estimating the TOAs and AOAs of a signal. Based on (11), we can estimate the TOAs and AOAs of incoming signals. Denote $(\widehat{\tau},\widehat{\theta})$ as the pair that maximizes the spectrum $S_i(\tau,\theta)$. We have

$$(\widehat{\tau},\widehat{\theta}) = \underset{\tau,\theta}{\text{argmax}}\ S_i(\tau,\theta). \tag{12}$$

Conventionally, the LoS-path-following signal has the strongest power; thus, on the receiver side, it is reasonable to consider that $(\widehat{\tau},\widehat{\theta})$ is the estimate of (τ_1,θ_1). Consequently, the distance between the transmitter and the receiver can be estimated as follows:

$$\widehat{d} = c\widehat{\tau}, \tag{13}$$

where c is the speed of light.

2) Trilateration-based Localization: We consider that localization is performed by the trilateration technique that measures the distances from the APs to a certain user.[2]

On the Cartesian plane, the coordinates of A_i ($i\in\{1,2,3\}$) and those of Bob are $(x_{\text{A}_i},y_{\text{A}_i})$ and $(x_{\text{B}},y_{\text{B}})$, respectively. Denote d_i as the *true* distance between A_i and B. Also, denote $\widehat{d}_i$ as the *estimated* distance between A_i and B. It is obvious that B is on the circle $(x_{\text{B}}-x_{\text{A}_i})^2+(y_{\text{B}}-y_{\text{A}_i})^2=d_i^2,\ i\in\{1,2,3\}$, and $(x_{\text{B}},y_{\text{B}})$ is the unique solution of the system of three equations. In practice, the true distance d_i is not attainable because the measurement of distance comes with errors. Thus, we will deal with the following system of three equations:

$$(x_{\text{B}}-x_{\text{A}_i})^2+(y_{\text{B}}-y_{\text{A}_i})^2=\widehat{d}_i^2,\ i\in\{1,2,3\}. \tag{14}$$

We can simply transform (14) into the following:

$$\mathbf{Q}\mathbf{z}=\mathbf{b}, \tag{15}$$

[2]The trilateration is based on TOAs. Apart from the trilateration technique, localization can also be performed by using hyperbolic curves, where hyperbolas are formed based on TDOAs instead of TOAs [25].

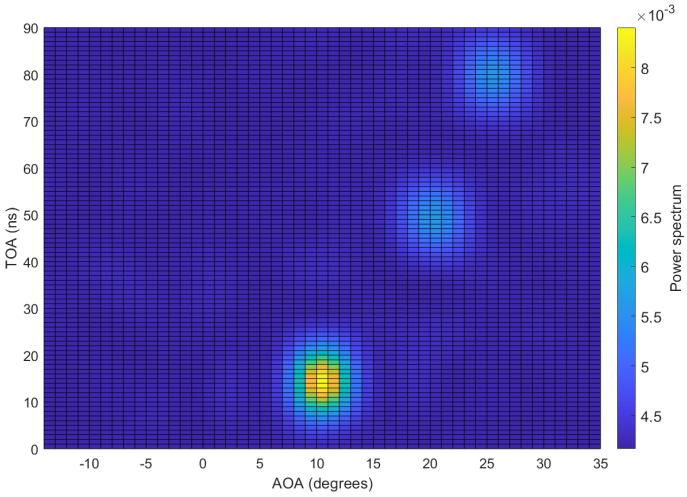


Fig. 2: An illustration of the MUSIC performance in estimating TOAs and AOAs at a certain AP i. The signal transmitted from a certain user u follows three paths (i.e, the LoS path and two NLOS paths) before reaching the i-th AP. Key parameters include $K_{\text{sub}} = 12$ sub-carriers, $f_c = 2.4$ GHz, $\Delta_F = 5$ MHz, $M = 10$ receive antennas, $\frac{P_1}{N_0} = 20$ dB, $\frac{P_2}{N_0} = \frac{P_3}{N_0} = 17$ dB. The The TOAs and AOAs are correctly estimated at $(\tau_{1i}, \theta_{1i}) = (15ns, 10^\circ)$, $(\tau_{2i}, \theta_{2i}) = (50ns, 20^\circ)$ and $(\tau_{3i}, \theta_{3i}) = (80ns, 25^\circ)$.

where

$$\mathbf{Q} \triangleq \begin{bmatrix} 2(x_{\text{A}_1} - x_{\text{A}_2}) & 2(y_{\text{A}_1} - y_{\text{A}_2}) \\ 2(x_{\text{A}_1} - x_{\text{A}_3}) & 2(y_{\text{A}_1} - y_{\text{A}_3}) \end{bmatrix} \in \mathbb{R}^{2\times 2},$$

$$\mathbf{b} \triangleq \begin{bmatrix} x_{\text{A}_1}^2 - x_{\text{A}_2}^2 + y_{\text{A}_1}^2 - y_{\text{A}_2}^2 + \widehat{d}_2^2 - \widehat{d}_1^2 \\ x_{\text{A}_1}^2 - x_{\text{A}_3}^2 + y_{\text{A}_1}^2 - y_{\text{A}_3}^2 + \widehat{d}_3^2 - \widehat{d}_1^2 \end{bmatrix} \in \mathbb{R}^{2\times 1},$$

$$\mathbf{z} = [x_{\text{B}}, y_{\text{B}}]^\top \in \mathbb{R}^{2\times 1}.$$

The equation (15) can be handled by using the method of least squares. A least-squares solution of $\mathbf{Qz} = \mathbf{b}$ is $\mathbf{z}^\star = \left(\mathbf{Q}^\dagger\mathbf{Q}\right)^{-1}\mathbf{Q}^\dagger\mathbf{b} \triangleq [\widehat{x}_{\text{B}}, \widehat{y}_{\text{B}}]^\top$. Finally, the estimated position of B is $(x, y) = (\widehat{x}_{\text{B}}, \widehat{y}_{\text{B}})$ in the Cartesian coordinate system.

Connecting the position estimation with the possibility of an attack, we can re-express $(\widehat{x}_{\text{B}}, \widehat{y}_{\text{B}})$ as follows:

$$(\widehat{x}_{\text{B}}, \widehat{y}_{\text{B}}) = \begin{cases} (\widehat{x}_{\text{B}|\mathcal{H}_0}, \widehat{y}_{\text{B}|\mathcal{H}_0}) & \text{under } (\mathcal{H}_0); \\ (\widehat{x}_{\text{B}|\mathcal{H}_1}, \widehat{y}_{\text{B}|\mathcal{H}_1}) & \text{under } (\mathcal{H}_1). \end{cases} \tag{16}$$

Herein, $(\widehat{x}_{\text{B}|\mathcal{H}_0}, \widehat{y}_{\text{B}|\mathcal{H}_0})$ implies the estimated position of B under the assumption that there is no influence from the attacker; whereas, $(\widehat{x}_{\text{B}|\mathcal{H}_1}, \widehat{y}_{\text{B}|\mathcal{H}_1})$ implies the estimated position of B under the influence of an attack. Fig. 4 illustrates the location estimation of a legitimate user under $\mathcal{H}_0$ and $\mathcal{H}_1$, respectively. It is shown that the trilateration-based localization method works well when there is no attack (i.e., under $\mathcal{H}_0$); however, the estimated position of the user is significantly far away from the actual position when there is an attack (i.e., under $\mathcal{H}_1$). It is thus necessary to detect if a received signal is affected by an attack.

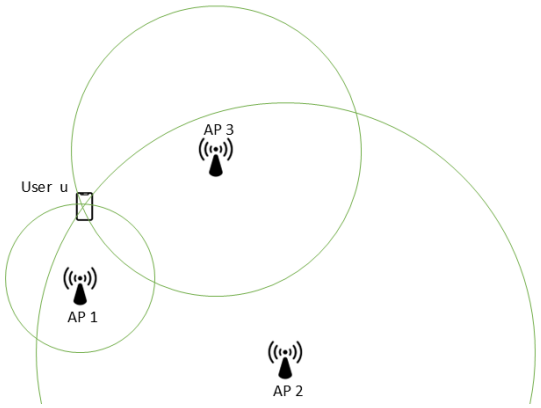


Fig. 3: An illustration of using the trilateration technique for localizing the position of a certain user u.

B. Multi-user Detection

In order to deal with multi-user interference, we will apply a multi-user detection (MUD) method based on zero-forcing (ZF) or minimum mean square error (MMSE) at each AP [26], [27]. In the case of MMSE, we calculate the *pseudo-inverse* of $\mathbf{H}_i$ as follows:[3]

$$\mathbf{G}^\star_{(\text{mmse},i)} = \left(\mathbf{H}_i^\dagger\mathbf{H}_i + \alpha\mathbf{I}\right)^{-1}\mathbf{H}_i^\dagger \triangleq \begin{bmatrix} \mathbf{G}^\star_{(\text{mmse},1\to i)} \\ \vdots \\ \mathbf{G}^\star_{(\text{mmse},U\to i)} \end{bmatrix} \tag{17}$$

Note that $\mathbf{G}^\star_{(\text{mmse},u\to i)} \in \mathbb{C}^{L_{\text{paths}}\times(K_{\text{sub}}M)}$ and $u \in \{1, \ldots, U\}$.

With small α, we have $\mathbf{G}^\star_{(\text{mmse},i)}\mathbf{H}_i \overset{\alpha\to 0}{\approx} \mathbf{I}$, which leads to the following:

$$\begin{bmatrix} \mathbf{G}^\star_{(\text{mmse},1\to i)} \\ \vdots \\ \mathbf{G}^\star_{(\text{mmse},U\to i)} \end{bmatrix} [\mathbf{H}_{1\to i}, \ldots, \mathbf{H}_{U\to i}] \approx \mathbf{I}$$

$$\Leftrightarrow \begin{cases} \mathbf{G}^\star_{(\text{mmse},u\to i)}\mathbf{H}_{u\to i} \approx \mathbf{I}; \\ \mathbf{G}^\star_{(\text{mmse},u\to i)}\mathbf{H}_{v\to i} \approx \mathbf{0}, \text{ for } u \neq v. \end{cases} \tag{18}$$

W.r.t. the u-th user, we calculate the following post-processing signal:

$$\begin{aligned} \widehat{\mathbf{s}}_{u\to i|\mathcal{H}_0} &= \mathbf{G}^\star_{(\text{mmse},u\to i)}\mathbf{r}_{i\mathcal{H}_0}(t) \\ &= \underbrace{\mathbf{G}^\star_{(\text{mmse},u\to i)}\mathbf{H}_{u\to i}}_{\approx \mathbf{I}}\mathbf{s}_u + \sum_{\substack{v=1 \\ v\neq u}}^{U} \underbrace{\mathbf{G}^\star_{(\text{mmse},u\to i)}\mathbf{H}_{v\to i}}_{\approx \mathbf{0}}\mathbf{s}_v \\ &\qquad + \mathbf{G}^\star_{(\text{mmse},u\to i)}\mathbf{n} \\ &\approx \mathbf{s}_u + \mathbf{G}^\star_{(\text{mmse},u\to i)}\mathbf{n}. \end{aligned} \tag{19}$$

From (19), we can also identify which user index is associated with the detected signal. Then, we can define the term $\mathbf{r}^{\text{pseudo}}_{u\to i|\mathcal{H}_0}(t) = \mathbf{H}_{u\to i}\widehat{\mathbf{s}}_{u\to i|\mathcal{H}_0} + \mathbf{n}(t)$ and find the pair $(\widehat{\tau}^{\text{pseudo}}_{u\to i}, \widehat{\theta}^{\text{pseudo}}_{u\to i})$ based on emulating the process described in Sub-section III-A1. The pair $(\widehat{\tau}^{\text{pseudo}}_{u\to i}, \widehat{\theta}^{\text{pseudo}}_{u\to i})$ can be understood as the (TOA, AOA) *fingerprints* of the u-th user on the MUSIC spectrum.

[3]The case of ZF corresponds to $\alpha = 0$.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3601022

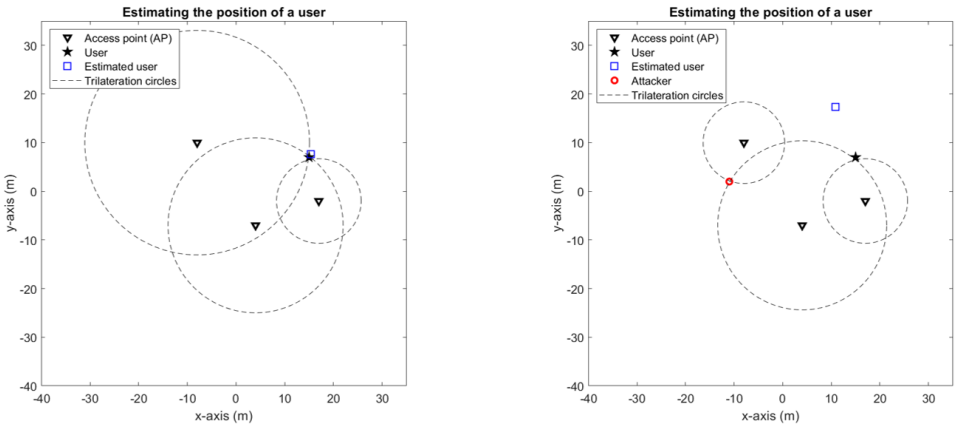


Fig. 4: Left-side sub-figure shows the position estimation of an STA (e.g., Bob 1) in the case of $\mathcal{H}_0$. Right-side sub-figures shows the the position estimation of Bob 1 in the case of $\mathcal{H}_1$.

C. Relationship between Localization and MUD in Phase (1)

The third sub-figure of Figure 1 depicts the proposed framework, where both localization and MUD are performed in Phase (1). Herein, MUD does not only help the APs detect the signals of different users, but also help the APs localize the positions of users. The role of MUD in improving localization is clarified in the sequel.

Firstly, as for localization, the APs use the trilateration technique to estimate the position of a certain user u. Noticeably, the trilateration technique relies on the estimation of distances from the APs to the u-th user. In the case of a single user, the localization method in Sub-section III-A1 can be directly applied, because the peaks on MUSIC spectra at the APs are associated the u-th user. However, in the case of multiple users, it is difficult to decide whether the peak of MUSIC spectrum at the i-th AP is associated with the u-th user or not. Thus, it is important to consider a *distance estimation process* (DEP) at each AP so that the impact of interference on MUSIC spectra can be reduced. Figure 5 depicts the DEP at the i-th AP. To differ the DEP at different APs, the DEP at the i-th AP will be named i-DEP. Note that i-DEP is itself *insufficient* for user positioning. To localize a certain user u, we need $\widehat{d}_{1u}$, $\widehat{d}_{2u}$ and $\widehat{d}_{3u}$. Herein, $\widehat{d}_{1u}$ is provided by 1-DEP at the 1-st AP, $\widehat{d}_{2u}$ is provided by 2-DEP at the 2-nd AP, and $\widehat{d}_{3u}$ is provided by 3-DEP at the 3-rd AP. This means that all the DEPs have to be performed by all the APs to provide necessary information for user positioning. The following flowchart depicts what has just been discussed:

$$\left.\begin{array}{l} \text{1-DEP} \rightarrow \widehat{d}_{1u} \\ \text{2-DEP} \rightarrow \widehat{d}_{2u} \\ \text{3-DEP} \rightarrow \widehat{d}_{3u} \end{array}\right\} \xrightarrow{\text{trilateration}} \text{position of user } u.$$

Secondly, since the presence of multiple users can cause confusion in estimating distances, it is necessary to eliminate the negative impact of multiple users to improve localization performance. Given that the DEP plays the most pivotal role in performing localization, we consider the elimination of multi-user impact in the DEP. As shown in Figure 5, i-DEP yields $\{\widehat{d}_{iu}, \ldots, \widehat{d}_{iU}\}$ in an iterative manner.[4] At each iteration, we update the MUSIC spectrum, determine the user index that is associated with the peak of the spectrum, and then estimate the distance based on the estimated TOA. As for determining the user index, there is a need to combine both MUSIC spectrum and MUD. To be more specific, at each iteration, after finding the (TOA, AOA) pair of the peak of the MUSIC spectrum, we compare this pair with the (TOA, AOA) fingerprints $(\widehat{\tau}_{u\rightarrow i}^{\text{pseudo}}, \widehat{\theta}_{u\rightarrow i}^{\text{pseudo}})$ (as described in Sub-section III-B) in order to determine which user is associated with the peak. Once the peak of the MUSIC spectrum has been determined to associate with a particular user u, we can then remove the contribution of the u-th user from the received signal $\mathbf{r}_i(t)$ in the next iteration. Denote $\check{\mathbf{r}}_i^{[q]}(t)$ as the remainder of the received signal $\mathbf{r}_i(t)$ after subtraction at the q-th iteration within i-DEP. Denote $\mathcal{U}_{\text{detected}}^{[q]}$ as the set of the indices of detected users. The cardinality of $\mathcal{U}_{\text{detected}}^{[q]}$ is $|\mathcal{U}_{\text{detected}}^{[q]}| = q-1$. If u is admitted to $\mathcal{U}_{\text{detected}}^{[q]}$ at the q-th iteration, this means that the u-th user is associated with the peak of MUSIC spectrum at the q-th iteration within i-DEP. Mathematically, $\check{\mathbf{r}}_i^{[q]}(t)$ is iteratively updated as follows:

$$\check{\mathbf{r}}_i^{[q]}(t) = \begin{cases} \mathbf{r}_i(t), & q = 1; \\ \mathbf{r}_i(t) - \sum_{u \in \mathcal{U}_{\text{detected}}^{[q]}} \mathbf{H}_{u\rightarrow i} \widehat{\mathbf{s}}_{u\rightarrow i|\mathcal{H}_0}(t), & q \geq 2. \end{cases} \tag{20}$$

By using $\check{\mathbf{r}}_i^{[q]}(t)$, we can first calculate the sample covariance $\check{\mathbf{C}}_i^{[q]}$ in a similar way to (10), and then calculate the spectrum $\check{S}_i^{[q]}(\tau, \theta)$ in a similar way to (11).

IV. Phases (2) and (3): ML-Aided Attacker Detection and Recovery of Reliable Localization

In this section, we present Phases (2) and (3) of the proposed framework. While Phase (2) employs ML to detect localization attacks, Phase (3) aims to restore the reliability of the localization process.

[4]The first estimated distance is not necessarily linked with $u = 1$ (i.e., User-1), because the (TOA, AOA) peak of the MUSIC spectrum in the 1-st iteration may be associated with another user (e.g., User-2). Also, the last estimated distance may not be necessarily linked with $u = U$.

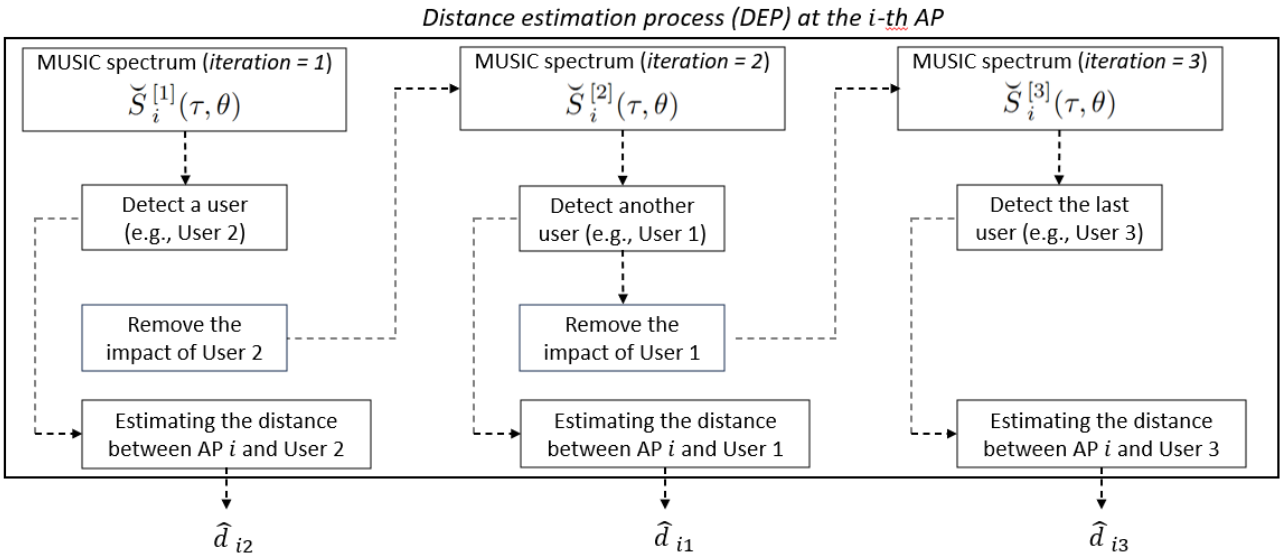


Fig. 5: The DEP is performed as part of the localization mechanism in Phase (1). The DEP, which is performed at the i-th AP, is called i-DEP. The principle of i-DEP is described in Sub-section III-C.

A. Phase (2): ML-Aided Attacker Detection

Anomaly detection models are trained on a training dataset (namely $\mathcal{T}^{\text{train}}$) before being tested on a separate testing dataset (namely, $\mathcal{T}^{\text{test}}$). Note that the training dataset $\mathcal{T}^{\text{train}} = \mathcal{T}^{\text{train}}_{\mathcal{H}_0}$ contains $|\mathcal{T}^{\text{train}}_{\mathcal{H}_0}|$ *legitimate* data points associated with the hypothesis $\mathcal{H}_0$ (i.e., non-attack behavior). Meanwhile, the testing dataset $\mathcal{T}^{\text{test}}_{\mathcal{H}_0} \cup \mathcal{T}^{\text{test}}_{\mathcal{H}_1}$ is the disjoint union of the set $\mathcal{T}^{\text{test}}_{\mathcal{H}_0}$ and the set $\mathcal{T}^{\text{test}}_{\mathcal{H}_1}$, i.e., $\mathcal{T}^{\text{test}} = \mathcal{T}^{\text{test}}_{\mathcal{H}_0} \cup \mathcal{T}^{\text{test}}_{\mathcal{H}_1}$, where the set $\mathcal{T}^{\text{test}}_{\mathcal{H}_0}$ contains $|\mathcal{T}^{\text{test}}_{\mathcal{H}_0}|$ data points associated with the hypothesis $\mathcal{H}_0$ and the set $\mathcal{T}^{\text{test}}_{\mathcal{H}_1}$ contains $|\mathcal{T}^{\text{test}}_{\mathcal{H}_1}|$ data points associated with the hypothesis $\mathcal{H}_1$. Since the detection performance depends on the data, it is important to form the structure of data for the learning purpose. For example, we have observed that the performance is not optimal if the training data uses $(\widehat{x}_{\text{B}|\mathcal{H}_0}, \widehat{y}_{\text{B}|\mathcal{H}_0})$ as features. On the other hand, the performance can be improved if the training data uses estimated distances and RSS values as described in Table III.

Moreover, since the detection is based on training the data with one label (i.e., $\mathcal{H}_0$), while the testing data contains both $\mathcal{H}_0$ and $\mathcal{H}_1$, we apply unsupervised learning. To be more specific, we will consider the following anomaly detection algorithms:

- One-class support vector machine (OC-SVM) is a special version of SVM, where a certain kernel function is employed for transforming the original feature space to a new separable feature space [28]–[30]. In the new feature space, a boundary that surrounds normal data points will be used as the decision function to identify the outliers outside the boundary. The training complexity is approximately $\mathcal{O}(|\mathcal{T}^{\text{train}}|^3)$; meanwhile, the testing complexity is $\mathcal{O}(|\mathcal{T}^{\text{test}}|n_{\text{SV}}n_{\text{dim}})$, where n_{SV} is the number of support vectors, and n_{dim} is the data dimension [31].
- Local outlier factor (LOF) belongs to the family of nearest-neighbor algorithms [28]–[30], [32]. In LOF, the distances from a data point to its nearest neighbors are calculated to determine the density of that point. By comparing the densities of data points, LOF can distinguish abnormal data points (i.e., outliers) from normal data points (i.e., inliers). The densities of outliers are lower than those of inliers. The training and testing complexities are $\mathcal{O}(|\mathcal{T}^{\text{train}}|^2 n_{\text{dim}})$ and $\mathcal{O}(|\mathcal{T}^{\text{test}}|^2 n_{\text{dim}})$, respectively. LOF is computationally expensive for large datasets [33].
- Isolation forest (iForest) generates random trees that partition the feature space and isolate individual instances [4], [29], [30], [34]. Unlike OC-SVM or LOF, iForest does not depend on learning a boundary or computing distances to detect anomalies. Instead, iForest computes the path lengths of trees and uses these values for scoring data points. By sorting the scores in descending order, the first top scores will be treated as outliers. The training complexity is approximately $\mathcal{O}(n_{\text{trees}}\psi \log \psi)$; meanwhile, the testing complexity is $\mathcal{O}(|\mathcal{T}^{\text{test}}|n_{\text{trees}} \log \psi)$, where n_{trees} is the number of trees and ψ is the sub-sampling size [35].
- Elliptic envelope (EE) hypothesizes that normal data points are drawn from a distribution [29], [36], [37]. Thus, EE estimates the covariance of the training data and models an ellipse so that the majority of normal training data points fit into the estimated ellipse. The Mahalanobis distance is then used as a metric for determining if a data point is an inlier or an outlier. The training complexity is approximately $\mathcal{O}\left(|\mathcal{T}^{\text{train}}|n^2_{\text{dim}} + n^3_{\text{dim}}\right)$; meanwhile, the testing complexity is $\mathcal{O}(|\mathcal{T}^{\text{test}}|n^2_{\text{dim}})$, suitable for moderate-sized datasets [38].

B. Phase (3): Recovery of Reliable Localization

In this section, we showcase that the use of a landmark can help the APs detect the attacker's position. Herein, a landmark is a node whose position is known to the APs. When being observed by the i-th AP, the set of TOAs of the landmark and the set of AOAs of the landmark are $\{\ddot{\tau}_{\ell,i}\}_{\ell=1}^{L_{\text{paths}}}$ and $\{\ddot{\theta}_{\ell,i}\}_{\ell=1}^{L_{\text{paths}}}$, respectively.

Scope	Index	Training data points	True labels	
Training dataset	1	$(\widehat{x}^{[1]}_{\mathrm{B}\|\mathcal{H}_0}, \widehat{y}^{[1]}_{\mathrm{B}\|\mathcal{H}_0})$	$(+1)$	No attack ($\mathcal{H}_0$)
	2	$(\widehat{x}^{[2]}_{\mathrm{B}\|\mathcal{H}_0}, \widehat{y}^{[2]}_{\mathrm{B}\|\mathcal{H}_0})$	$(+1)$	
	$\vdots$	$\vdots$	$\vdots$	
	$T^{\text{train}}_{\mathcal{H}_0}$	$(\widehat{x}^{[T^{\text{train}}_{\mathcal{H}_0}]}_{\mathrm{B}\|\mathcal{H}_0}, \widehat{y}^{[T^{\text{train}}_{\mathcal{H}_0}]}_{\mathrm{B}\|\mathcal{H}_0})$	$(+1)$	

TABLE I: The structure of the training dataset with 2 features.

Scope		Index	Testing data points	True labels	
Testing dataset	In the set $\mathcal{T}^{\text{test}}_{\mathcal{H}_0}$	1	$(\widehat{x}^{[1]}_{\mathrm{B}\|\mathcal{H}_0}, \widehat{y}^{[1]}_{\mathrm{B}\|\mathcal{H}_0})$	$(+1)$	No attack ($\mathcal{H}_0$)
		2	$(\widehat{x}^{[2]}_{\mathrm{B}\|\mathcal{H}_0}, \widehat{y}^{[2]}_{\mathrm{B}\|\mathcal{H}_0})$	$(+1)$	
		$\vdots$	$\vdots$	$\vdots$	
		$T^{\text{test}}_{\mathcal{H}_0}$	$(\widehat{x}^{[T^{\text{test}}_{\mathcal{H}_0}]}_{\mathrm{B}\|\mathcal{H}_0}, \widehat{y}^{[T^{\text{test}}_{\mathcal{H}_0}]}_{\mathrm{B}\|\mathcal{H}_0})$	$(+1)$	
	In the set $\mathcal{T}^{\text{test}}_{\mathcal{H}_1}$	1	$(\widehat{x}^{[1]}_{\mathrm{B}\|\mathcal{H}_1}, \widehat{y}^{[1]}_{\mathrm{B}\|\mathcal{H}_1})$	(-1)	Attack ($\mathcal{H}_1$)
		2	$(\widehat{x}^{[2]}_{\mathrm{B}\|\mathcal{H}_1}, \widehat{y}^{[2]}_{\mathrm{B}\|\mathcal{H}_1})$	(-1)	
		$\vdots$	$\vdots$	$\vdots$	
		$T^{\text{test}}_{\mathcal{H}_1}$	$(\widehat{x}^{[T^{\text{test}}_{\mathcal{H}_1}]}_{\mathrm{B}\|\mathcal{H}_1}, \widehat{y}^{[T^{\text{test}}_{\mathcal{H}_1}]}_{\mathrm{B}\|\mathcal{H}_1})$	(-1)	

TABLE II: The structure of the testing dataset with 2 features.

When the landmark communicates with the i-th AP, the received signal vector at the i-th AP is denoted as $\ddot{\mathbf{r}}_i(t) = \ddot{\mathbf{r}}_{u\to i|\mathcal{H}_0}(t)$ in the case of no attack or $\ddot{\mathbf{r}}_i(t) = \ddot{\mathbf{r}}_{u\to i|\mathcal{H}_1}(t)$ in the case of an attack. The expressions of $\ddot{\mathbf{r}}_{u\to i|\mathcal{H}_0}(t)$ and $\ddot{\mathbf{r}}_{u\to i|\mathcal{H}_1}(t)$ are the same as those of $\mathbf{r}_{u\to i|\mathcal{H}_0}(t)$ and $\mathbf{r}_{u\to i|\mathcal{H}_1}(t)$ in (3) and (4). Since the landmark is known to all the APs, it is reasonable to consider that the APs know about the landmark's TOAs and AOAs. This means that the i-th AP has the information about $\{\ddot{\tau}_{1,i}, \ldots, \ddot{\tau}_{\ddot{L}_{\text{paths}},i}\}$ and $\{\ddot{\theta}_{1,i}, \ldots, \ddot{\theta}_{\ddot{L}_{\text{paths}},i}\}$, where $\ddot{L}_{\text{paths}}$ is the number of paths that can be estimated/recorded by an AP. We have $\ddot{L}_{\text{paths}} \leq L_{\text{paths}}$. For example, a signal from the landmark arrives at an AP following $L_{\text{paths}} = 100$ paths; however, only $\ddot{L}_{\text{paths}} = 5$ paths have significant contribution in the received signal and can be recognized by the AP through a localization method like the MUSIC spectrum.

As discussed in the previous section, ML algorithms have been applied to detect whether the received signal is associated with an attack or not. Since tracing a target by using ToA/TDoA is possible [39], [40], we consider localizing the detected attacker with the aid of landmarks in this section.

Recall that the user moves around the landmarks. When the positions of landmarks are known to the APs, the i-th AP can keep a record of the TOAs and the AOAs $\{\ddot{\theta}_{\ell,i}\}_{\ell=1}^{L_{\text{paths}}}$ of the landmark. The received signal vector at the i-th AP is denoted as $\ddot{\mathbf{r}}_i(t) = \ddot{\mathbf{r}}_{i\mathcal{H}_0}(t)$ in the case of no attack or $\ddot{\mathbf{r}}_i(t) = \ddot{\mathbf{r}}_{i\mathcal{H}_1}(t)$ in the case of an attack. The expressions of $\ddot{\mathbf{r}}_{i\mathcal{H}_0}(t)$ and $\ddot{\mathbf{r}}_{i\mathcal{H}_1}(t)$ are the same as those of $\mathbf{r}_{i\mathcal{H}_0}(t)$ and $\mathbf{r}_{i\mathcal{H}_1}(t)$ in (3) and (4). Since the landmark is known to all the APs, it is reasonable to consider that the i-th AP has the information about the landmark's TOAs $\{\ddot{\tau}_{1,i}, \ldots, \ddot{\tau}_{\ddot{L}_{\text{paths}},i}\}$ and AOAs $\{\ddot{\theta}_{1,i}, \ldots, \ddot{\theta}_{\ddot{L}_{\text{paths}},i}\}$, where $\ddot{L}_{\text{paths}}$ is the number of paths that can be estimated/recorded by the i-th AP.

Once an attack has been correctly detected by an ML algorithm, the next goal is to extract the attacker's position from the received signal $\ddot{\mathbf{r}}_{i\mathcal{H}_1}(t)$. Since $\ddot{\mathbf{r}}_{i\mathcal{H}_1}(t)$ includes both the contributions of the landmark (i.e., the contributions from L_{paths} paths) and those of the attacker (i.e., the contributions from L_{attacker} paths). Meanwhile, the landmark's contributions w.r.t. $\ddot{L}_{\text{paths}}$ has been recognized. Thus, the i-th AP can remove the $\ddot{L}_{\text{paths}}$ contributions of the landmark from the received signal as follows:

$$\ddot{\mathbf{r}}_{\mathcal{H}_1,\text{ext}} = \ddot{\mathbf{r}}_{i\mathcal{H}_1} - \underbrace{\mathbf{a}(\ddot{\tau}_{1,i}, \ddot{\theta}_{1,i})\ddot{s}_1(t)e^{-j2\pi f_c \ddot{\tau}_{1,i}}}_{\text{the contribution of the landmark's LoS path}}. \quad (21)$$

An important note from (21) is that the extracted signal $\ddot{\mathbf{r}}_{\mathcal{H}_1,\text{ext}}$ may be mostly constituted by the contributions of attacker. Thus, the MUSIC spectrum based on $\ddot{\mathbf{r}}_{\mathcal{H}_1,\text{ext}}$ will be likely to result in estimating the TOAs and AOAs of the attacker, because the $\ddot{L}_{\text{paths}}$ biggest contributions of landmark have been previously removed. Since the user is close to one of the landmarks, the user's TOA and AOA values are also close to those of one of the landmarks. If the user is close to one landmark instead of another, the subtraction in (21) will result in the attenuation of a peak that is close to the location of the landmark's peak of the MUSIC spectrum. Therefore, by performing the subtraction for different landmarks and then comparing the MUSIC spectra, we can determine which landmark the user is closest to. Finally, the highest peak of the MUSIC spectrum after subtraction will be determined to be associated with the attacker.

V. Numerical Results and Discussions

This section presents numerical results. We consider a scenario with $U = 3$ users. The geometric setup of the APs, the users and the attacker are as follows: The APs are located at $(4, -7)$, $(-8, 10)$, and $(17, -2)$, respectively. The positions of users B_1, B_2, and B_3 are uniformly distributed within circles of radius 6 meters centered at the landmarks $(15, 7)$, $(21, 3)$, and $(11, 5)$, respectively. Meanwhile, the position of the attacker is uniformly distributed within a circle of radius 6 meters centered at $(-11, 2)$.

As for the path loss, we use the free-space path loss model. Then, the path loss between a transmitter (e.g., the user) and the i-th AP can be modelled as $\mathfrak{L}_i^{(\ell)} = (4\pi d_i^{(\ell)})^2/(G_{\text{dir}}\lambda^2)$, where G_{dir} is the directivity, and $d_i^{(\ell)}$ is the distance traveled when the signal follows the ℓ-th path. For simplicity, we set $G_{\text{dir}} = 0.01$. In the case of the LoS path (i.e., $\ell = 1$), $d_i^{(1)}$ is the *Euclidean* distance between the transmitter and the i-th AP. Note that the symbol d_i defined in Section III-A2 is exactly the same as $d_i^{(1)}$. In the case of NLoS paths (i.e., $\ell > 1$), we always have $d_i^{(\ell)} > d_i^{(1)}$. In simulations, we consider a scenario where each user has $L_{\text{paths}} = 3$ paths, and the attacker has $L_{\text{jam}} = 3$ paths. Each user has a transmit power of $P_{\text{Tx}} = 1$ mW, and the attacker has the transmit power of $P_{\text{jam}} = 1$ mW. As for the noise, we model the noise variance as $N_0 = k_{\text{Boltzmann}} T_{\text{Kelvin}} \text{BW}$, where $k_{\text{Boltzmann}} = 1.38 \times 10^{-23}$ is the Boltzmann constant, T_{Kelvin} is the temperature in Kelvin, and BW is the bandwidth. Regarding the MMSE detectors at the APs, we set $\alpha = 0.1$. Regarding the MUSIC spectrum,

Scope	Index	Training data points	True labels	
Training data	1	$(\widehat{d}^{[1]}_{1j\|\mathcal{H}_0}, \widehat{d}^{[1]}_{2j\|\mathcal{H}_0}, \widehat{d}^{[1]}_{3j\|\mathcal{H}_0}, \text{RSS}^{[1]}_{1j\|\mathcal{H}_0}, \text{RSS}^{[1]}_{2j\|\mathcal{H}_0}, \text{RSS}^{[1]}_{3j\|\mathcal{H}_0})$	$(+1)$	No attack
	$\vdots$	$\vdots$	$\vdots$	
	$T^{\text{train}}_{\mathcal{H}_0}$	$(\widehat{d}^{[T^{\text{train}}_{\mathcal{H}_0}]}_{1j\|\mathcal{H}_0}, \widehat{d}^{[T^{\text{train}}_{\mathcal{H}_0}]}_{2j\|\mathcal{H}_0}, \widehat{d}^{[T^{\text{train}}_{\mathcal{H}_0}]}_{3j\|\mathcal{H}_0}, \text{RSS}^{[T^{\text{train}}_{\mathcal{H}_0}]}_{1j\|\mathcal{H}_0}, \text{RSS}^{[T^{\text{train}}_{\mathcal{H}_0}]}_{2j\|\mathcal{H}_0}, \text{RSS}^{[T^{\text{train}}_{\mathcal{H}_0}]}_{3j\|\mathcal{H}_0})$	$(+1)$	

TABLE III: The structure of the training dataset with 6 features.

Scope		Index	Testing data points	True labels	
Testing data	in $\mathcal{T}^{\text{test}}_{\mathcal{H}_0}$	1	$(\widehat{d}^{[1]}_{1j\|\mathcal{H}_0}, \widehat{d}^{[1]}_{2j\|\mathcal{H}_0}, \widehat{d}^{[1]}_{3j\|\mathcal{H}_0}, \text{RSS}^{[1]}_{1j\|\mathcal{H}_0}, \text{RSS}^{[1]}_{2j\|\mathcal{H}_0}, \text{RSS}^{[1]}_{3j\|\mathcal{H}_0})$	$(+1)$	No attack
		$\vdots$	$\vdots$	$\vdots$	
		$T^{\text{test}}_{\mathcal{H}_0}$	$(\widehat{d}^{[T^{\text{test}}_{\mathcal{H}_0}]}_{1j\|\mathcal{H}_0}, \widehat{d}^{[T^{\text{test}}_{\mathcal{H}_0}]}_{2j\|\mathcal{H}_0}, \widehat{d}^{[T^{\text{test}}_{\mathcal{H}_0}]}_{3j\|\mathcal{H}_0}, \text{RSS}^{[T^{\text{test}}_{\mathcal{H}_0}]}_{1j\|\mathcal{H}_0}, \text{RSS}^{[T^{\text{test}}_{\mathcal{H}_0}]}_{2j\|\mathcal{H}_0}, \text{RSS}^{[T^{\text{test}}_{\mathcal{H}_0}]}_{3j\|\mathcal{H}_0})$	$(+1)$	
	in $\mathcal{T}^{\text{test}}_{\mathcal{H}_1}$	1	$(\widehat{d}^{[1]}_{1j\|\mathcal{H}_1}, \widehat{d}^{[1]}_{2j\|\mathcal{H}_1}, \widehat{d}^{[1]}_{3j\|\mathcal{H}_1}, \text{RSS}^{[1]}_{1j\|\mathcal{H}_1}, \text{RSS}^{[1]}_{2j\|\mathcal{H}_1}, \text{RSS}^{[1]}_{3j\|\mathcal{H}_1})$	(-1)	Attack
		$\vdots$	$\vdots$	$\vdots$	
		$T^{\text{test}}_{\mathcal{H}_1}$	$(\widehat{d}^{[T^{\text{test}}_{\mathcal{H}_1}]}_{1j\|\mathcal{H}_1}, \widehat{d}^{[T^{\text{test}}_{\mathcal{H}_1}]}_{2j\|\mathcal{H}_1}, \widehat{d}^{[T^{\text{test}}_{\mathcal{H}_0}]}_{3j\|\mathcal{H}_1}, \text{RSS}^{[T^{\text{test}}_{\mathcal{H}_1}]}_{1j\|\mathcal{H}_1}, \text{RSS}^{[T^{\text{test}}_{\mathcal{H}_1}]}_{2j\|\mathcal{H}_1}, \text{RSS}^{[T^{\text{test}}_{\mathcal{H}_1}]}_{3j\|\mathcal{H}_1})$	(-1)	

TABLE IV: The structure of the testing dataset with 6 features.

$T_{\text{samples}} = 100$ samples are used for peak finding, with a TOA resolution of 10^{-9} s and an AOA resolution of $(\pi/180)$ radians. Unless stated otherwise, other system parameters are as follows: $P_{\text{Tx}} = 1$ mW; $P_{\text{jam}} = 1$ mW; $M = 10$ antennas; $K_{\text{sub}} = 64$ subcarriers; $f_c = 2.4$ GHz; $\Delta_F = 5$ MHz.

As for the unsupervised learning algorithms, their hyper-parameters are as follows. For iForest, LOF, and EE, we set the proportion of outliers, namely the "contamination" parameter, to 0.01. Meanwhile, for OC-SVM, we set a similar parameter, namely ν, to 0.01, where ν serves as an upper bound on the fraction of anomalies. Additionally, other hyper-parameters are set as follows: i) For iForest, the parameter controlling randomness in feature selection is set to 30; ii) For LOF, the number of neighbors is set to 30; iii) For OC-SVM, we use the widely-adopted RBF kernel function, with the kernel coefficient equal to the inverse of the product between the number of features (i.e., n_{dim}) and the data variance; and iv) For EE, the parameter controlling randomness in subsampling is set to 30.

A. Performance of Anomaly Detection models

After training an anomaly detection model on a training dataset, the trained model will be evaluated by a testing dataset. Thus, the term "performance" in this section implies the performance in the testing phase but not the training phase. We observe that there are *significant* performance differences when the datasets have different structures. We first present an analysis using a data set with two features (the Cartesian coordinates) and then show how considering other features such as received signal strengths and distances to APs can drastically improve the performance.

1) Datasets with 2 features: We first examine the datasets with 2 features, as described in Tables I and II. These features are the Cartesian coordinates. The distribution of data points on the two-dimensional plane is visualized in Fig. 6, where $(\widehat{x}, \widehat{y})$ is the estimated position of B.

Fig. 7 illustrates the learning curves of the four ML models after the training process. The learning curves are formed on

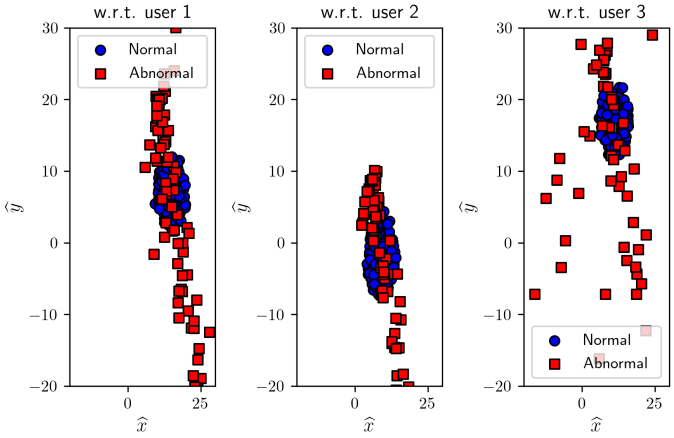


Fig. 6: Visualization of data points in the space with the 2 features.

the basis that most training data points lie inside the curves. In the testing phase, any data point inside the curves will be predicted as a normal data point (i.e., which is associated with $\mathcal{H}_0$), while any data point outside the curves will be predicted as an *abnormal* data point (i.e., which is associated with $\mathcal{H}_1$). Based on the prediction, we can compute the four following basic values:

- True *positive* (TP): The number of actual normal data points that are correctly predicted as "*normal*".
- False *negative* (FN): The number of actual normal data points that are incorrectly predicted as "*abnormal*".
- False *positive* (FP): The number of actual abnormal data points that are incorrectly predicted as "*normal*".
- True *negative* (TN): The number of actual abnormal data points that are correctly predicted as "*abnormal*".

From TP, FN, FP and TN, the sensitivity (i.e., the true positive rate) and the specificity (i.e., the true negative rate) are calculated. Fig. 8 shows the sensitivity and specificity of four ML models. We can observe that the sensitivity of each ML model is as high as approximately 99%; however,

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3601022

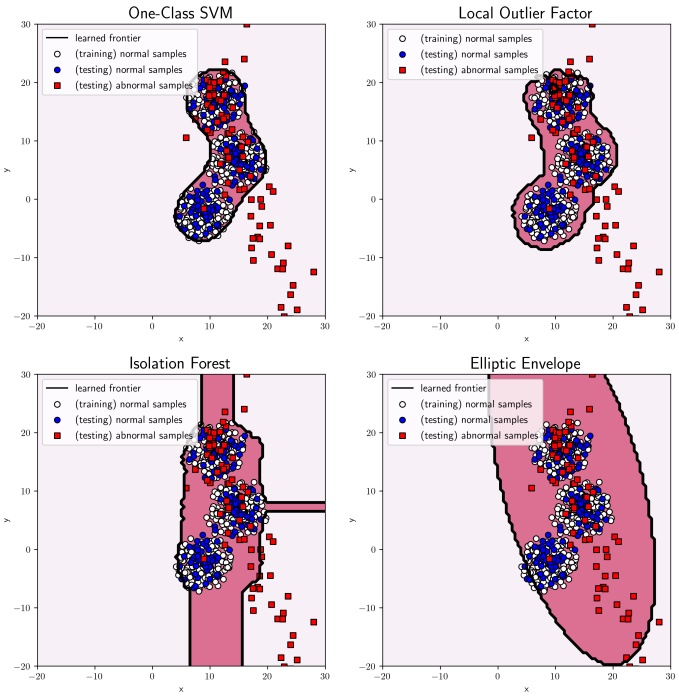


Fig. 7: Illustrating the learning boundaries when algorithms are trained on the datasets with only 2 features.

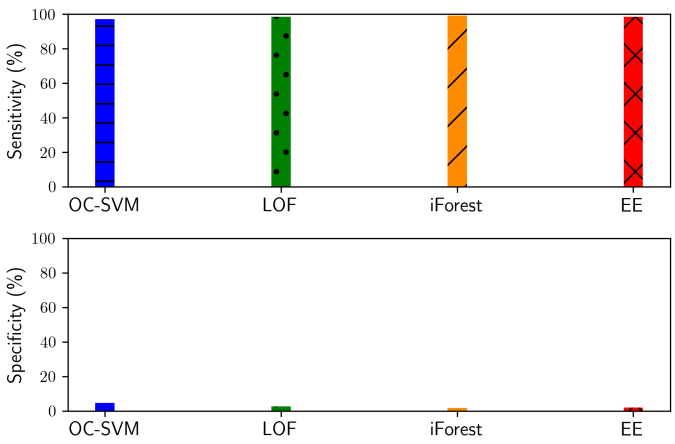


Fig. 8: The sensitivity and specificity of 4 ML algorithms are depicted, given that the datasets have 2 features.

the specificity is less than 7%. This observation implies that the four trained ML models miss at least 93% of actual abnormal data points, because the specificity is the indicator for identifying the actual abnormal data points. In terms of security, not being able to detect 93% of all attacks is severe, regardless of the high rate of detecting normal points.

Fig. 9 shows the receiver operating characteristic (ROC), which is another performance evaluation for binary classifiers when considering thresholds. The ROC depicts the true positive rate (i.e., the sensitivity) against the false positive rate (i.e., 1− specificity). For example, if we require the false positive rate at most 30% (i.e., 0.3), then the true positive rate can be 30%. In general, the true positive rate is almost the same as

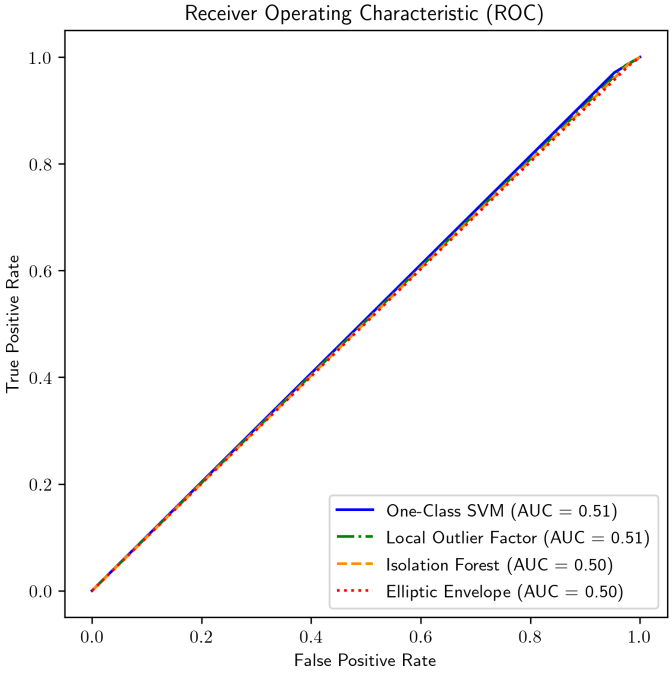


Fig. 9: Receiver operating characteristics (ROCs) of 4 algorithms.

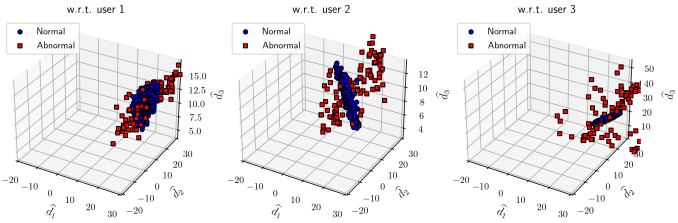


Fig. 10: Visualization of data points in the space with the first 3 features.

the false positive rate, thus leading to the area under the curve (AUC) is almost 0.5. This means that all the 4 ML models do not perform well when being trained on the data with 2 features (i.e., the x- and y- coordinates). Obviously, using statistical knowledge of positions is inefficient for detecting PHY localization attacks.

2) Datasets with 6 features: Now, we consider the datasets with 6 features, as described in Tables III and IV. These features include the estimated distances and the received signal strengths (RSSs) at the APs, instead of the estimated positions obtained by the localization process. Fig. 10 visualizes the data points in the three-dimensional space whose axes correspond to the first three features.

In Fig. 11, the sensitivity and specificity of 4 ML models are shown. While the all ML models have high sensitivity (i.e., above 85%), the specificity considerably varies with different ML models. We can see that the specificity of OC-SVM or LOF is quite high value (i.e., above 85%), Isolation Forest has moderate value (i.e., around 70%), but Elliptic Envelope has very low specificity. Comparing the specificity in Fig. 11 with that in Fig. 8, we observe an improvement for OC-SVM,

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3601022

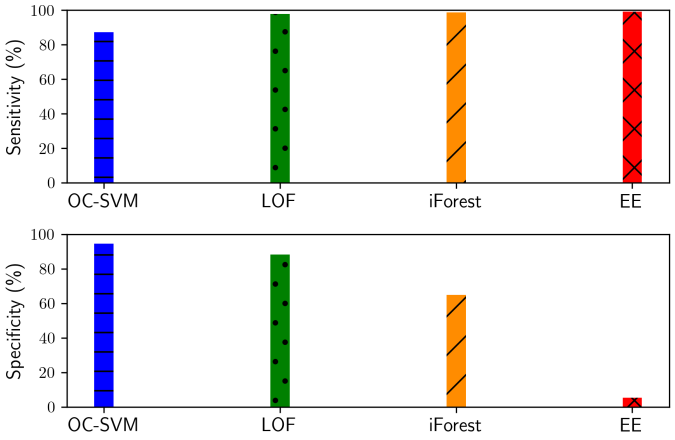


Fig. 11: The sensitivity and specificity of 4 ML algorithms are depicted, given that the datasets have 6 features.

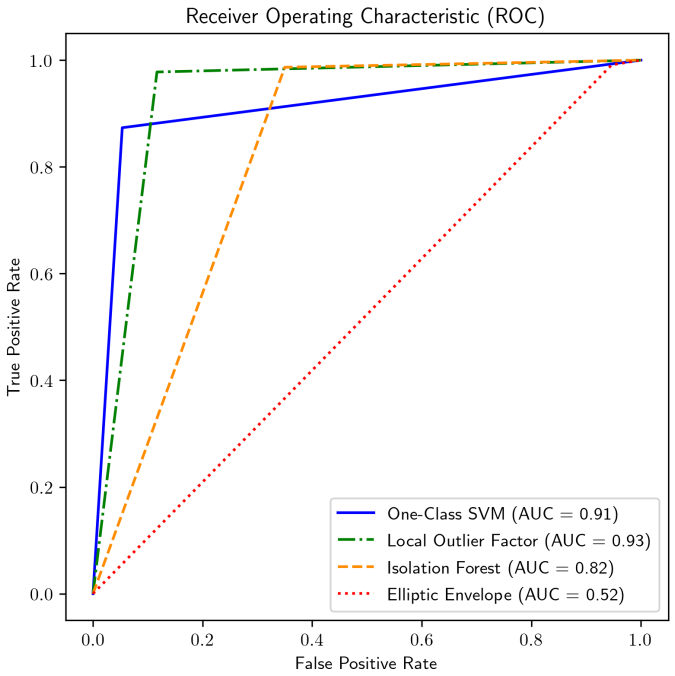


Fig. 12: Receiver operating characteristics (ROCs) of 4 algorithms.

LOF and Isolation Forest. Noticeably, the performance of OC-SVM and that of LOF are sufficiently high in terms of both sensitivity and specificity.

Fig. 12 shows the ROC in the case the data has 6 features. It shows that OC-SVM and LOF are the best classifiers because their area under curves (AUCs) are highest. For example, if we require the false positive rate at most 11%, then OC-SVM and LOF classifiers can satisfy the requirement. Compared with the results in Fig. 9, it is clear that the usage of datasets with 6 features is much better than the usage of datasets with only 2 features, because we can train OC-SVM and LOF models to create classifiers with high sensitivity and specificity.

Figure 13 presents the sensitivities and specificities of the four algorithms under different attacker power levels (i.e.

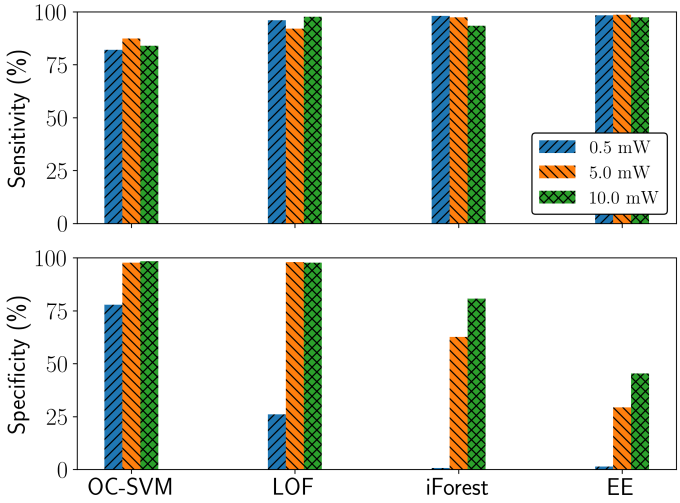


Fig. 13: The sensitivities and specificities of 4 algorithms are evaluated under different attacker power levels.

$P_{\text{jam}} = \{0.5, 5, 10\}$ mW). At low attacker power, OC-SVM achieves moderate sensitivity and specificity, both above 75%. By contrast, LOF, iForest, and EE exhibit moderate-to-high sensitivity but very poor specificity. As the attacker power increases, sensitivity remains high across all algorithms, but specificity improves significantly, especially for LOF and iForest. Meanwhile, EE maintains high sensitivity, but its specificity remains low, with only modest improvement as the attacker power increases. Overall, OC-SVM appears robust and stable; meanwhile, the other algorithms tend to over-detect anomalies by misclassifying normal samples as malicious, especially at lower attacker power levels.

B. Performance of Localizing the Attacker

The results in this subsection are obtained using the proposed framework. Without it, estimating the positions of legitimate users under attack is not possible. Instead, localization performance degrades, as shown in Figure 4. Based on the successful detection of an attack, the position of the attacker will be estimated. The estimation performance are presented in Figures 14 and 15.

To be more specific, Figure 14 compares the actual and estimated distances from each AP to the attacker, given that the Rician factor is equal to 10, the user's power and the attacker's power are equal to 1.5 mW and 1 mW, respectively. It can be seen that from the perspective of each AP, the difference between the true distance and estimated one is insignificant. This figure confirms again that each AP can relatively estimate the distance from it to the attacker, thus confirming the efficiency of the proposed method in Section IV. Moreover, this observation means that when all APs coordinate to perform the trilateration based on the estimated distances (as described in Section III-A2), the spatial position of the attacker can be estimated with a high accuracy.

Figure 15 shows the mean absolute error between the true position of the attacker and the estimated one, where different environmental settings are considered via the Rician factor.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3601022

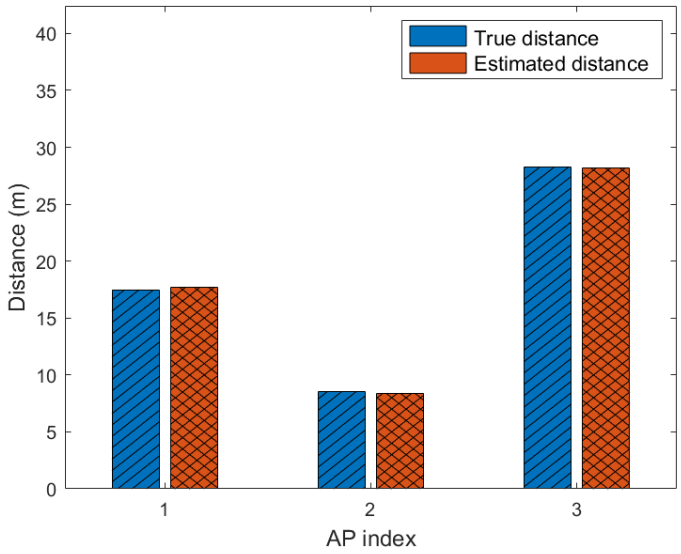


Fig. 14: From the perspective of each gNB, the true position of the attacker and the estimated one are compared.

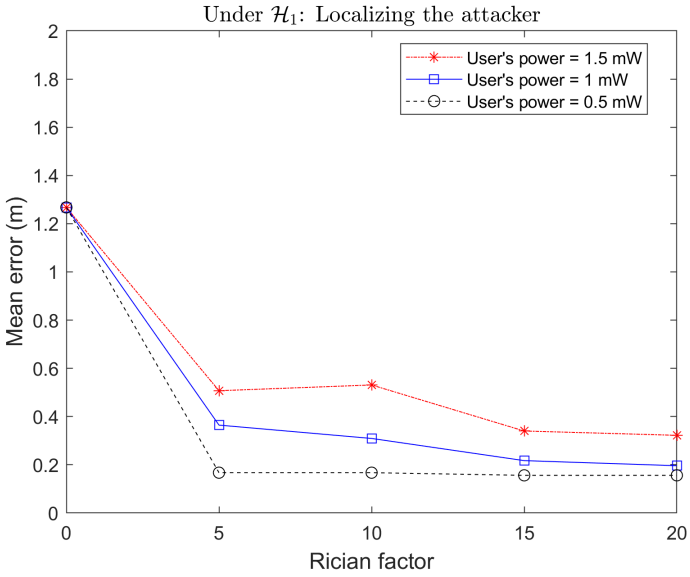


Fig. 15: The absolute mean error between the true position of the attacker and the estimated one is depicted against the Rician factor.

Herein, the Rician factor is calculated as the ratio of the LoS-path-following signal's power to the total power of other NLoS-path-following signals. When the Rician factor is equal to zero (i.e., there is no LoS path), the error is at the peak of around 1.25 meters. However, when the Rician factor increases (i.e., the LoS component becomes stronger), the error gradually reduces. It is noticeable that the error also depends on the transmit power, meaning that the possibility of detecting the attacker depends on the user's power. While the attacker's power is assumed to be 1 mW, the user's power varies among $\{0.5, 1, 1.5\}$ mW. If the user transmits at lower power, the contribution of the user in the received signal is smaller; thus making the contribution of the attacker becomes higher and making the attacker to be more easily detected. Consequently,

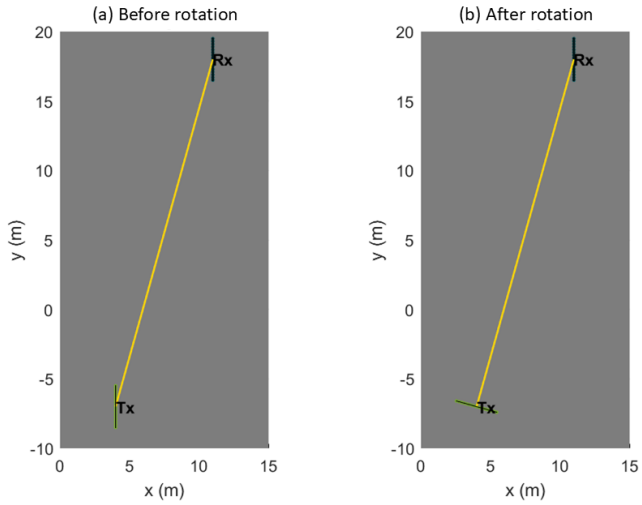


Fig. 16: An illustration of the transmit antenna array before and after rotation.

the performance in the case of $P_{\text{UE}} = 0.5$ mW is better than that in the two other cases.

C. Position-Aware Payload Transmission in Phase (4)

Based on the basis that the positions of nodes have been estimated, we demonstrate the benefits of the position-aware payload transmission in the downlink through simulations. Indeed, it is not necessary to consider all possible transmissions from a certain AP to a certain user. For demonstration purposes only, we consider the payload transmission from the 1-st AP to the 3-rd user, as shown in Figures 16–17. Moreover, the results in those figures can be viewed as an example of improving the LBS for the user.

To be more specific, Figure 16 depicts the arrangement of the transmit antenna array before and after rotation. The original arrangement of the transmit antenna array is parallel to the y-axis. After rotation, the transmit antenna array points toward the position of the receive antenna array for better performance. The difference in performance is illustrated in the next figure.

In Figure 17, we evaluate the PER at the user in case of not using a movable-antenna array and in the case of using it. Since the use of the movable-antenna array allows for rotating the direction of the beam in a desired direction, there is a difference in the performance before and after rotation. When not using the movable-antenna array (i.e., not rotating the antenna array), the best beam has ID $= 7$, which corresponds to the solid curve in Figure 17. By contrast, when using the movable-antenna array (i.e., after rotating it), the best beam has ID $= 4$, which corresponds to the dashed curve. Obviously, the use of the movable-antenna array leads to a lower PER at the user. This observation also promotes the idea of integrating movable-antenna arrays with position-aware transmissions to enhance LBSs.

In Figure 18, we show the packet error rate (PER) at the user of interest (i.e., the 3-rd user) for all 7 possible beam patterns (i.e., beam IDs), given that each beam ID corresponds

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2025.3601022

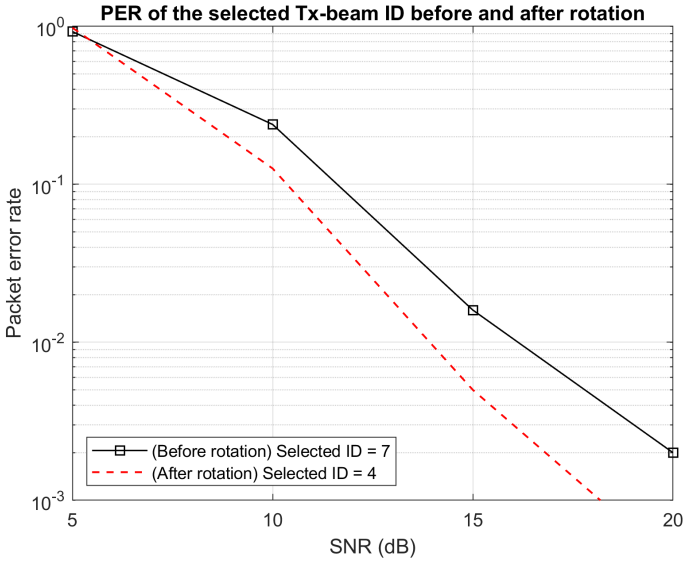


Fig. 17: The PER w.r.t. the selected Tx-beam ID is depicted for two cases: i) before rotation; and ii) after rotation.

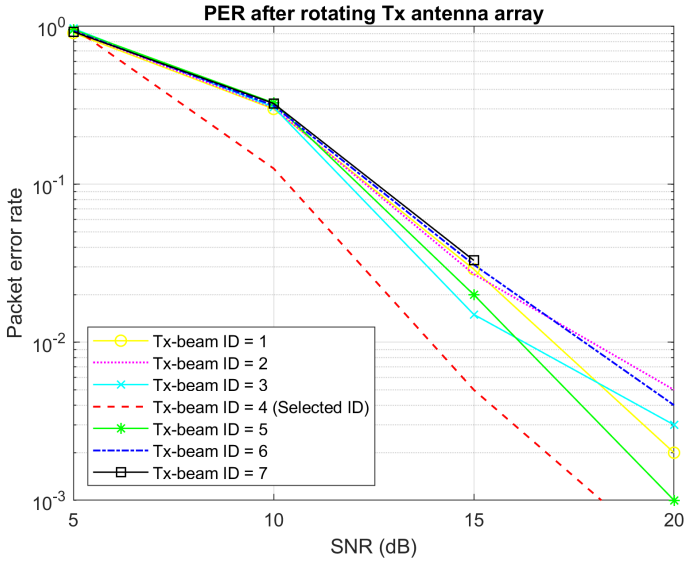


Fig. 18: The PER is depicted for the case that rotation is performed. The selected beam ID is the 4-th ID.

to a different choice of beam ID. Based on the knowledge of the estimated positions of the user and the attacker, the best beam ID is selected so that its main lobe does not point toward the attacker but points toward the user. We can see that the selected beam ID (i.e., ID $=$ 4) yields the best PER for the user. Moreover, when the AP is equipped with a movable-antenna array, the selected beam ID can be fine-tuned so that the direction of its main lobe is the same as the geometric direction of the AP-user link. Since the selection of the best beam ID depends on the localization results in the proposed framework, Figure 18 also demonstrates its effectiveness in achieving optimal PER. In contrast, without the proposed framework, performance degrades due to an inaccurate localization process.

D. More Discussions

The proposed framework is inherently scalable and generalizable. Although we adopt a minimal setting with three access points in a two-dimensional plane for conceptual clarity, the same trilateration method can be directly extended to support three-dimensional localization when more access points are available, such as four or more. In dense urban environments where many APs are typically present, the system can select the most suitable subset based on criteria such as signal strength or geometric configuration. Furthermore, the framework accommodates multiple users by incorporating the MUD module that is capable of distinguishing signals from different users. Once separated, the trilateration method can be applied to estimate the position of each user individually. The framework has been evaluated with three users and can be extended to more users as long as the MUD component successfully associates the measurements with the correct user. These features confirm that the proposed method is well suited to realistic deployment scenarios, without requiring fundamental modifications to the core framework.

VI. Conclusion

In this paper, we proposed a framework that reciprocally combines both localization and detection to deal with PHY attacks and restore the reliability of localization. On the one hand, during the localization process, a novel TOA-based distance estimation method was proposed, where the MUD and MUSIC techniques are used for sequentially detecting users and estimating related distances. Moreover, the data gleaned from the localization process can also be used as the input data for the anomaly detection algorithms. On the other hand, the results of the attacker detection process are used for restoring the reliability of the localization process by removing the attacker's impact. Finally, as for the downlink, we showed the benefits of using corrected positions for improving the secure transmission with the help of movable-antenna arrays.

References

[1] F. Campolo, A. Blaga, M. Rea, A. Lozano, and X. Costa-Perez, "5GNSS: Fusion of 5G-NR and GNSS localization for enhanced positioning accuracy and reliability," *IEEE Trans. on Veh. Tech.*, vol. 73, no. 9, pp. 13 558–13 568, 2024.

[2] M. D. Redzic, C. Laoudias, and I. Kyriakides, "Image and WLAN bimodal integration for indoor user localization," *IEEE Trans. on Mobile Comp.*, vol. 19, no. 5, pp. 1109–1122, 2020.

[3] C. Zhao, H. Du, D. Niyato, J. Kang, Z. Xiong, D. I. Kim, X. Shen, and K. B. Letaief, "Generative AI for secure physical layer communications: A survey," *IEEE Trans. on Cog. Commun. and Netw.*, pp. 1–24, 2024.

[4] T. M. Hoang, A. Vahid, H. D. Tuan, and L. Hanzo, "Physical layer authentication and security design in the machine learning era," *IEEE Commun. Sur. & Tut.*, vol. 26, no. 3, pp. 1830–1860, 2024.

[5] C. Wu, X. Yi, W. Wang, L. You, Q. Huang, X. Gao, and Q. Liu, "Learning to localize: A 3D CNN approach to user positioning in massive MIMO-OFDM systems," *IEEE Transactions on Wireless Communications*, vol. 20, no. 7, pp. 4556–4570, 2021.

[6] Y. Teng, P. Zhang, X. Chen, X. Jiang, and F. Xiao, "Phy-layer authentication exploiting channel sparsity in mmwave MIMO UAV-ground systems," *IEEE Trans. on Info. Foren. and Secu.*, vol. 19, pp. 4642–4657, 2024.

[7] M. Kazemian, T. Dagiuklas, and J. Jasperneite, "Direction estimation of the attacked signal in PBCH of 5G NR," *IEEE Commun. Lett.*, vol. 28, no. 7, pp. 1639–1643, 2024.

[8] J. He, M. Niu, P. Zhang, and C. Qin, "Enhancing PHY-layer authentication in RIS-assisted iot systems with cascaded channel features," *IEEE Internet of Things Journal*, vol. 11, no. 14, pp. 24 984–24 997, 2024.

[9] Y. Yang, J. Li, N. Luo, Z. Yan, Y. Zhang, and K. Zeng, "BatchAuth: A physical layer batch authentication scheme for multiple backscatter devices," *IEEE Trans. on Info. Foren. and Secu.*, vol. 19, pp. 9452–9466, 2024.

[10] M. Niu, P. Zhang, J. He, Y. Zhang, and Z. Liu, "PHY-layer authentication exploiting spatial channel and radiometric signatures for mmWave MIMO systems," *IEEE Commun. Lett.*, pp. 1–5, 2025.

[11] Y. Lin, S. Jin, M. Matthaiou, and X. You, "Channel estimation and user localization for IRS-assisted MIMO-OFDM systems," *IEEE Transactions on Wireless Communications*, vol. 21, no. 4, pp. 2320–2335, 2022.

[12] Z. Yu, X. Hu, C. Liu, M. Peng, and C. Zhong, "Location sensing and beamforming design for IRS-enabled multi-user ISAC systems," *IEEE Transactions on Signal Processing*, vol. 70, pp. 5178–5193, 2022.

[13] Y. Lu, O. Kaltiokallio, M. Koivisto, J. Talvitie, E. S. Lohan, H. Wymeersch, and M. Valkama, "Bayesian filtering for joint multi-user positioning, synchronization and anchor state calibration," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 8, pp. 10 949–10 964, 2023.

[14] R. Wang, Z. Xing, E. Liu, and J. Wu, "Joint localization and communication study for intelligent reflecting surface aided wireless communication system," *IEEE Transactions on Communications*, vol. 71, no. 5, pp. 3024–3042, 2023.

[15] E. Y. Menta, N. Malm, R. Jantti, K. Ruttik, M. Costa, and K. Leppanen, "On the performance of AoA-based localization in 5G ultra–dense networks," *IEEE Access*, vol. 7, pp. 33 870–33 880, 2019.

[16] A. Mohammadi, M. Rahmati, and H. Malik, "Location-aware beamforming for MIMO-enabled UAV communications: An unknown input observer approach," *IEEE Sensors Journal*, vol. 22, no. 8, pp. 8206–8215, 2022.

[17] Y. Li, J. Yang, S.-L. Shih, W.-T. Shih, C.-K. Wen, and S. Jin, "Efficient iot devices localization through Wi-Fi CSI feature fusion and anomaly detection," *IEEE Internet of Things J.*, pp. 1–17, 2024.

[18] J. Yuan, Y. Cai, Y. Chen, N. Xie, P. Zhang, L. Huang, and D. Niyato, "Efficient detection of cooperative external attacks in wireless localization systems," *IEEE Trans. on Wireless Commun.*, pp. 1–17, 2024.

[19] M. Beko and S. Tomic, "Toward secure localization in randomly deployed wireless networks," *IEEE Internet of Things Journal*, vol. 8, no. 24, pp. 17 436–17 448, 2021.

[20] Q. Cheng, Y. Zhou, H. Liu, L. Yang, Z. Ma, and P. Fan, "Physical layer authentication in UAV communications with channel randomness and jamming uncertainty," *IEEE Trans. on Veh. Tech.*, pp. 1–6, 2025.

[21] P. Zhang, K. Han, Y. Zhang, Y. Shen, F. Xiao, and X. Jiang, "Distributed physical layer authentication framework exploiting array pattern feature for mmWave MIMO systems," *IEEE Trans. on Mobile Comp.*, vol. 24, no. 7, pp. 6430–6445, 2025.

[22] R. Li, H. Hu, and Q. Ye, "RFTrack: Stealthy location inference and tracking attack on Wi-Fi devices," *IEEE Trans. on Info. Foren. and Sec.*, vol. 19, pp. 5925–5939, 2024.

[23] Y. Zheng, M. Sheng, J. Liu, and J. Li, "Exploiting AoA estimation accuracy for indoor localization: A weighted AoA-based approach," *IEEE Wirel. Commun. Lett.*, vol. 8, no. 1, pp. 65–68, 2019.

[24] T. M. Hoang, T. van Chien, T. van Luong, S. Chatzinotas, B. Ottersten, and L. Hanzo, "Detection of spoofing attacks in aeronautical ad-hoc networks using deep autoencoders," *IEEE Trans. on Info. Foren. and Secu.*, vol. 17, pp. 1010–1023, 2022.

[25] Y. Wang and K. C. Ho, "Unified near-field and far-field localization for AOA and hybrid AOA-TDOA positionings," *IEEE Trans. on Wirel. Commun.*, vol. 17, no. 2, pp. 1242–1254, 2018.

[26] K. Zu, J. Zhu, and M. Haardt, "Uplink multi-user MIMO detection via parallel access," in *ICASSP 2019 - 2019 IEEE Int. Conf. on Acous., Speech and Sig. Process. (ICASSP)*, 2019, pp. 4365–4369.

[27] S. Sowmya, G. Muthukrishnan, and K. Giridhar, "Low-complexity linear decoupling of users for uplink massive MU-MIMO detection," in *2024 IEEE 99th Veh. Tech. Conf. (VTC2024-Spring)*, 2024, pp. 1–6.

[28] P. Martins, A. B. Reis, P. Salvador, and S. Sargento, "Physical layer anomaly detection mechanisms in IoT networks," in *IEEE/IFIP Network Operations and Management Symposium (NOMS)*, 2020, pp. 1–9.

[29] S. Shriram and E. Sivasankar, "Anomaly detection on shuttle data using unsupervised learning techniques," in *2019 Int. Conf. on Compu. Intel. and Knowl. Economy (ICCIKE)*, 2019, pp. 221–225.

[30] Y. Yengi, A. Kavak, and H. Arslan, "Physical layer detection of malicious relays in LTE-A network using unsupervised learning," *IEEE Access*, vol. 8, pp. 154 713–154 726, 2020.

[31] S. Kang, D. Kim, and S. Cho, "Approximate training of one-class support vector machines using expected margin," *Computers Industrial Engineering, Elsevier*, vol. 130, p. 772–778, 2019.

[32] G. Cerar, H. Yetgin, B. Bertalanic, and C. Fortuna, "Learning to detect anomalous wireless links in IoT networks," *IEEE Access*, vol. 8, pp. 212 130–212 155, 2020.

[33] M. Goldstein, "FastLOF: An expectation-maximization based local outlier detection al- gorithm," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, Japan*, 2012, pp. 2282–2285.

[34] J. Lesouple, C. Baudoin, M. Spigai, and J.-Y. Tourneret, "Generalized isolation forest for anomaly detection," *Pattern Recognition Letters*, vol. 149, pp. 109–119, 2021.

[35] F. T. Liu, K. M. Ting, and Z. H. Zhou, "Isolation forest," in *Eighth IEEE International Conference on Data Mining, Pisa, Italy*, 2008, pp. 413–422.

[36] M. Ashrafuzzaman, S. Das, A. A. Jillepalli, Y. Chakhchoukh, and F. T. Sheldon, "Elliptic envelope based detection of stealthy false data injection attacks in smart grid control systems," in *2020 IEEE Symp. Series on Comp. Int. (SSCI)*, 2020, pp. 1131–1137.

[37] M. Antonini, M. Vecchio, F. Antonelli, P. Ducange, and C. Perera, "Smart audio sensors in the internet of things edge for anomaly detection," *IEEE Access*, vol. 6, pp. 67 594–67 610, 2018.

[38] P. J. Rousseeuw and K. van Driesse, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, p. 212–223, 1999.

[39] F. Ricciato, S. Sciancalepore, and G. Boggia, "Tracing a linearly moving node from asynchronous time-of-arrival measurements," *IEEE Commun. Lett.*, vol. 20, no. 9, pp. 1836–1839, 2016.

[40] P. Wang, Z. Yan, and K. Zeng, "BCAuth: Physical layer enhanced authentication and attack tracing for backscatter communications," *IEEE Trans. on Info. Foren. and Secu.*, vol. 17, pp. 2818–2834, 2022.

Tiep M. Hoang received the Ph.D. degree in Electronics, Electrical Engineering and Computer Science from Queen's University Belfast, UK, in 2019. From 2020 to 2022, he was a Postdoctoral Fellow at University of Southampton, UK. From 5/2022 to 8/2023, he was a Postdoctoral Fellow at University of Colorado Denver, US. He is currently a Postdoctoral Fellow at Rochester Institute of Technology, US. His research interests include wireless communications, physical-layer security, convex optimization, and machine learning.

Alireza Vahid (Senior Member, IEEE) received the B.Sc. degree in electrical engineering from Sharif University of Technology, Tehran, Iran, in 2009, and the M.Sc. and Ph.D. degrees in electrical and computer engineering from Cornell University, Ithaca, NY, USA, in 2012 and 2015 respectively. From 2015 to 2017, he worked as a postdoctoral research scientist at the Information Initiative at Duke University, Durham, NC, USA. From 2017 to 2023, he was an Assistant Professor of electrical engineering at the University of Colorado at Denver, Denver, CO, USA. He is currently a Gleason Endowed Associate Professor of Electrical and Microelectronic Engineering at Rochester Institute of Technology, Rochester, NY, USA. His research interests include network information theory, wireless communications, and applications of coding theory in high-performance computer memory systems. He received the 2015 Outstanding Ph.D. Thesis Research Award, the 2010 Director's Ph.D. Teaching Award, Jacobs Scholar Fellowship in 2009 from Cornell University, a 2013 Qualcomm Innovation Fellowship, a 2019 Lab Venture Challenge Award, and a 2021 SONY Faculty Innovation Award. He currently serves as an associate editor for IEEE Communications Letters and IEEE Transactions on Information Theory.