RBS-MLP: A Hybrid Computational Intelligence Framework for Rogue Base Stations Detection in 5G Mobile Networks

Journal:	*IEEE Transactions on Emerging Topics in Computational Intelligence*
Manuscript ID	TETCI-2025-2810
Manuscript Type:	Full Paper
Keywords:	5G mobile networks, computational intelligence, machine learning, and rogue base station detection

RBS-MLP: A Hybrid Computational Intelligence Framework for Rogue Base Stations Detection in 5G Mobile Networks

***Abstract*—Rogue Base Station (RBS) is a persistent security threat in 5G and 5G beyond networks, which exploits radio access procedures to attract user equipment through abnormal signal behavior. This paper introduces RBS-MLP, a hybrid Computational Intelligence (CI) framework that integrates data-driven learning with state-based reasoning to safely detect RBSs early and reliably. Unlike existing client- or cloud-based solutions, RBS-MLP operates entirely within the network, enabling finer-grained detection without requiring modifications to user equipment. This framework incorporates novel engineered temporal signal features, including the Rate of Change of Received Signal (RoCH), and a probation-based finite state machine to capture both short-term dynamics and long-term consistency in base station behavior. Under realistic mobility scenarios, a multilayer perceptron classifier is trained and tested on time-series measurement reports across multiple generated datasets and signal window sizes to identify legitimate entities and rogue entities. In this study, the proposed approach is evaluated by analyzing more than 200,000 measurements collected in a large-scale 5G vehicle platooning scenario. Experimental results demonstrate detection accuracy of up to 99.9%, with consistently low false positive rates while maintaining low computational overhead suitable for real-time deployment. RBS-MLP provides a scalable, efficient, and standards-aligned baseline to identify RBS in 5G intelligent transportation systems environments. In addition to being compliant with 3GPP Release 18, this framework is compatible with emerging RAN architectures, making it suitable for 5G services that require low latency.**

***Index Terms*—5G mobile networks, computational intelligence, machine learning, and rogue base station detection.**

I. Introduction

THE intelligence and autonomy of 5G-enabled systems have expanded the attack surface in mobile networks, particularly at the radio access layer [1]. Rogue base stations (RBS) that impersonate legitimate network infrastructure to manipulate radio procedures remain a persistent and evolving threat, despite recent security improvements in 5G standards [2], [3]. Identifying adversarial activities demands not only accurate classification but also intelligent analysis of temporal signal dynamics [4], [5]. An RBS attack can happen during the initial cell search stage in the 5G New Radio (NR) when a User Equipment (UE) looks for a suitable BS to camp on. During this stage, the UE listens to the wireless broadcast channel for the Synchronizing Signal (SS) from nearby BSs. Next, the UE selects a BS based on the received SSs and initiates a wireless connection. If an RBS broadcasts a spoofing SS with high Received Signal Strength (RSS) during the cell search mechanism, the UE may be enticed to it and try to camp on it rather than any legitimate BS [6]. While UE refers broadly to mobile phones, laptops, or vehicles, our experiments focus on autonomous vehicles in a platooning use case. However, the proposed RBS-MLP model is applicable to other UE types in mobile networks. In the emerging 5G world, it will be vital for infrastructure providers to protect against such attacks to secure the communications platform and protect client data and identity.

Since the inception of early GSM networks, RBS attacks have continuously evolved and persisted. These can be categorized as denial-of-service (DoS) attacks on mobile devices or networks, provision of fraudulent services, and compromise of subscribers' privacy. The impact of these attacks varies greatly among cellular network generations but remains significant owing to the multiple interconnections across a diverse set of current and legacy networks [2]. The development of 5G communications has already led to some advancements in RBS detection, such as Subscription Permanent Identifier (SUPI) concealment, guaranteed Globally Unique Temporary Identity (GUTI) refreshment, and protected redirections [3]. At the same time, other security mechanisms inherited from previous generations include mutual authentication between UE and network, secure algorithm negotiations, and integrity-protected signalling [7]. Despite these advances, the current position is that 5G remains vulnerable to RBS attack [3].

Most RBS detection systems analyze data gathered from UE in one of the following three ways: (i) locally on the UE itself, (ii) over the cloud, or (iii) through the broader network between UE and Cloud [8]–[11]. Of these, the first group is prone to false positives because a UE cannot understand the complete status of the network view at any given time. In addition, UE-side detection systems often need software updates on the device or root privileges, which are uncommon and may be difficult for some users. In the second group, the devices send their Measurement Report (MR) [12] to a central server in the 5G Core for analysis; otherwise, they operate on the same premise as UE-side detectors and, as a result, suffer from the same efficacy and scalability concerns. The third group, Network-based detection systems, are projected to perform better in analysis since, unlike UEs, mobile networks know the system's global status. However, network-based detection requires additional infrastructure to collect data across multiple locations and protocols. In this study, we focus on the third group, which is more applicable to 5G and cellular systems.

In recent years, there has been considerable interest in Machine Learning (ML) for detecting security threats. ML models can identify malicious threats by learning the behavior of both attack and legitimate scenarios. ML-based classification systems can provide high levels of precision in the identification of potential aggressors [13], [14].

From a Computational Intelligence (CI) viewpoint, RBS detection is a challenge involving non-stationary time-series data, partial observability, and real-time decision making under

0000–0000/00$00.00 © 2021 IEEE

uncertainty. Traditional rule- and threshold-based methods struggle to adapt to dynamic radio conditions, while purely data-driven models often lack contextual reasoning for safe handovers. This motivates the need for hybrid frameworks combining learning-based inference with explicit state-aware reasoning [15].

This paper proposes RBS-MLP, a hybrid CI framework that combines supervised learning with symbolic state-transition models to detect RBS in 5G networks. We implement a probation-based Finite State Machine (FSM) embedding learned signal intelligence to enable accurate and early RBS identification before handover [16]. This CI framework is evaluated in a vehicular platooning scenario; however, it can be adapted to other CI-driven anomaly detection and applied to a wide range of 5G-enabled systems, including smartphones, IoT devices, and pedestrian terminals [17].

An CI approach is presented here for detecting RBS in 5G networks. Unlike prior methods that rely on heuristic rules, device-rooted apps, or static thresholds, RBS-MLP integrates signal behaviour learning, state transition modeling, and realistic simulation to achieve robust, scalable detection. The main contributions are as follows:

- A hybrid CI framework that incorporates supervised learning and symbolic state-transition reasoning to identify RBS.
- A new temporal feature abstraction, Rate of Change of Received Signal Strength (RoCH), captures dynamic signals that are difficult to model using static thresholds or RSS values.
- An early exclusion of suspected RBS through probation-based FSM embedded into the network decision process.
- An integrated network-based detection mechanism that does not require any UE-side software changes or additional hardware, ensuring scalability and standards compliance.
- A redesigned handover mechanism to integrate a trust evaluation component that enables the network to block suspected RBS during the measurement report stage, before connection establishment, thus mitigating attack impact.
- Simulate realistic highway conditions by deploying a vehicle platooning scenario involving over 200,000 timestamped MR records, incorporating a combination of legitimate and rogue BS deployments.

The rest of the paper is structured as follows. In Section II, we review the related research in RBS detection systems while identifying their limitations. Section III presents a novel RBS detection model comprising a pre-processing component and a decision maker. Section IV proposes the RBS-MLP model. Section V presents the performance evaluation and results for various case studies based on our realistic dataset of radio information and RSS measurements taken by a simulated vehicle traveling along various sections of a road. Finally, the conclusions and future works are discussed in Section VI.

II. Related Work

This section reviews existing RBS detection systems, mentioned in the introduction, along with their capabilities, limitations, and drawbacks.

The UE side includes client-side applications that perform the identification within the UEs. This includes mobile phones, vehicles, IoT devices [18], [19], etc. Android IMSI-Catcher Detector (AIMSICD) [8], Cell Spy Catcher [20], Catcher-Catcher [21], and SnoopSnitch [9] are some applications that fall into this group. To provide some level of protection, these applications require high privileges and low-level access to baseband chips to reach their full potential. Even though Cell Spy Catcher and AIMSICD results have not been persuasive, Cell Spy Catcher can at least be used to determine if the local network figures have been modified. SnoopSnitch seems to be the most advanced of the alternatives, as it reliably informs the user immediately after the threat is detected. In contrast, Cell Spy Catcher only provides a warning and associated information. SnoopSnitch, on the other hand, only works on Qualcomm-based Android phones and requires root access. Similarly, CatcherCatcher attempts to detect RBS activity by detecting irregularities in mobile networks, but it only works on Osmocom phones. To summarize, these applications are still in the early stages of development for detecting RBS attacks. They have a lower detection rate, generate more false positives, and require unusually high-level access, making them unsuitable for the general public [22].

Techniques for cloud-based detection are based on analyzing the crowdsourced data from a large number of UEs to detect and geolocate RBS units. FBS-Radar [10], a large-scale RBS detection and location system, identifies an RBS by automating the collection of suspicious SMS messages from end-user devices. In addition, these reports, including RSS, cell identifier, and UE MAC addresses, are sent to a server to analyze and evaluate different techniques that exploit this data to identify RBS installations accurately without analyzing the content of the SMS messages. Van Do et al. [23] suggested a methodology for detecting abnormal behaviour from an RBS in public data sets using ML approaches. In [24], the experiment was extended using ML and exploiting a signature-based strategy with characteristics such as location and the relationship between the identification number of the UE and subscription. These investigations used a publicly available data set from Aftenposten [25] to demonstrate the utility of ML approaches, but with the drawback that UEs must report their measurements to a server on the cloud for analysis; otherwise, they operate like a client-side detector. As a result, they suffer from effectiveness and scalability issues.

Network-based detection techniques analyze the core of the cellular network. In [11], a technique for IMSI catcher detection has been proposed that uses existing operational data from the mobile network used in the mobility management of mobile stations. The MRs delivered by the UEs to BSs containing information on the cell and surrounding cells are used to detect IMSI catchers. Regarding analysis, network-based detection systems are projected to outperform client-side detectors since mobile networks know the system's global status, unlike UEs. However, a limitation with [11] is that they only cover 2G radio access technology. Murat [2] is a network-based approach for recognising RBSs on several 3GPP Radio Access Technologies (RAT) without changing

mobile phones or monitoring equipment. Murat employs the global state to include information on all connected mobile phones, the mobile network state, and its deployment and setup history; it outperforms earlier systems.

The RBS-MLP framework uses a hybrid CI methodology, which combines symbolic reasoning and ML. This paradigm addresses the contextual limitations of purely data-driven models, enhancing both adaptability and trustworthiness [26]. Such hybrid frameworks—combining learning with rule-based systems—are proven to offer superior real-time accuracy against complex threats like rogue base stations [27].

However, most existing network-based approaches rely on static rule-based detection or heuristic thresholds, which limit their ability to adapt to the highly dynamic radio conditions present in mobile networks. Furthermore, only a few methods explicitly address the time-sensitive nature of vehicular mobility and handover decisions in 5G NR networks, which is critical for secure management services of vehicular networks.

In contrast to existing techniques, this work proposes an integrated network-side detection framework that combines ML-driven dynamic signal modelling with a FSM–based trust model, specifically tailored to meet the low-latency, high-mobility requirements of intelligent transportation systems. Unlike recent deep learning approaches such as FBSDetector [28], which target general cellular networks, our solution is designed for seamless integration with 3GPP-compliant RAN architectures and optimized for real-time RBS detection in vehicular environments.

III. Proposed Detection Model for Rogue Base Stations

In this section, first, (i) the framework of the detection model is proposed, and (ii) then a novel handover protocol will provide a key feature for the ML method to build the main component of the RBS detection system.

The analyzer component of the RBS-MLP is an ML-based approach based on MR; it can be considered an additional feature for Murat [2]. Murat offers a network-based approach for identifying RBS that run on any 3GPP radio access technology without needing to modify mobile phones. However, the analyzer of Murat consists of data processors and Rule-based methods; we analyze the MR using ML methods [2], [29].

A. Proposed Architecture for RBS detection

Fig. 1 describes the proposed architecture to model an RBS identification system to detect rogue agents using ML methods. The UE in RRC_CONNECTED mode builds the MR based on signals received from gNBs currently in range and sends it to the 5G Radio Access Network (RAN). The proposed system performs the data analysis, identifies suspected RBS and eliminates them from consideration for handover [30]. As a result, the suspected RBS is never included in the version of the MR used to assess the need for a handover event. The BS identifies the need for a handover and, if required, initiates the protocol. Upon detection of an RBS, the network operators can be informed so that legal action and other post-incident activities can be initiated. For example, they can alarm the UE from camping on the RBS [31], [32].

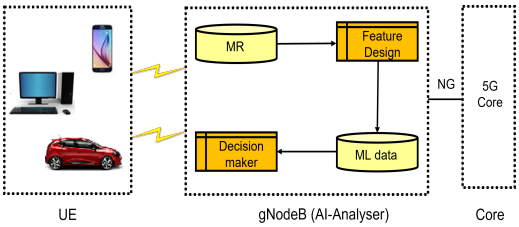

Fig. 1. RBS detector architecture, illustrating the flow of MR from UE to the 5G RAN for analysis and classification using AI.

Such analyzers might be included in either the 5G RAN or the 5G Core, however, in this study, the integration of the analysis into the 5G RAN is examined for scalability, which is most readily done when the RBS detector is situated at the point where the gNodeB receives the MR data [2]. The components of the AI Analyser in Fig. 1, comprise a Feature Design element, an AI-based Decision-maker element, and two MR and ML data storage units. Feature Design aims to create informative and relevant input features that help the model distinguish between the two classes. The ML approaches will be used in the decision-making function. The Decision Maker, as a significant component of the AI Analyser, will receive the ML data and apply the ML method to identify rogues. Other analyzer functions can employ various strategies for classification, but we will demonstrate the effectiveness of an ML approach. The sections that follow expand on the architecture's description and details.

B. Handover Process

In a 5G environment, an IoT device typically has a selection of BS units within its reception range, and the way the device decides which BS to attach to is based on an analysis of the device's MR. The MR identifies the most vital received BS signals from the current location and orientation and is updated periodically, usually every second. Depending on the RSS of the currently connected BS and the signal strengths of alternate candidate BS units, the device will be told by the network either to stay with the current BS or handover to a more robust alternative. Handover management is critical for ensuring that UEs may move freely between cells while still receiving high-quality communication services. The gNB is in charge of managing UE migration across cells. Typically the handover decision in 5G RAN is based on the MR produced by UEs [33]. The handover process is that the UE is currently connected to a gNB (the serving BS); if there is another BS in the MR with a level that exceeds the threshold value, then we handover to it. Fig. 2 illustrates the Received Signal Strength (RSS) from a collection of 30 BS encountered by a platoon leader along a stretch of urban motorway. As the UE moves along the highway, the handover process is activated, resulting in the connected signal strength profile illustrated in Fig. 3.

C. Feature Design for RBS Detection

Legitimate BSs typically adjust power gradually due to coverage management policies and UE handover thresholds,

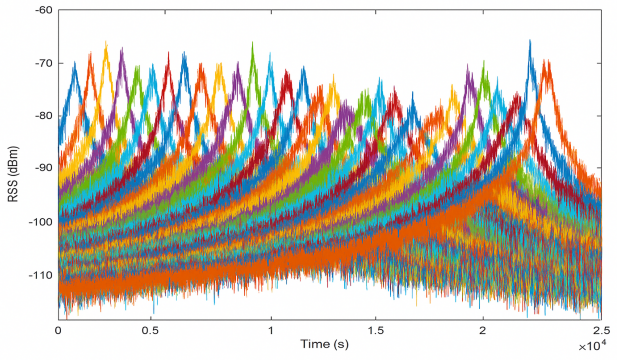

Fig. 2. RSS of base stations encountered by a vehicle along a highway in a legitimate scenario (each colour illustrates the RSS from different BS).

while RBS units may abruptly change power levels to attract UEs. A steep increase and fall in RSS is a characteristic of the profile of a typical rogue, because RBSs are often configured to transmit at unusually high power levels for short durations to attract nearby UEs and override legitimate base stations [12], which can be used as a fingerprint for detection. Therefore, this section defines and implements two new features, including the RoCH and probation period. These informative features are calculated and monitored for all BS and RBS that the UE detects. The main idea is that the RoCH is a better indicator for RBS detection than "raw" RSS data, as RBS tends to increase at a more rapid rate than "regular" BS.

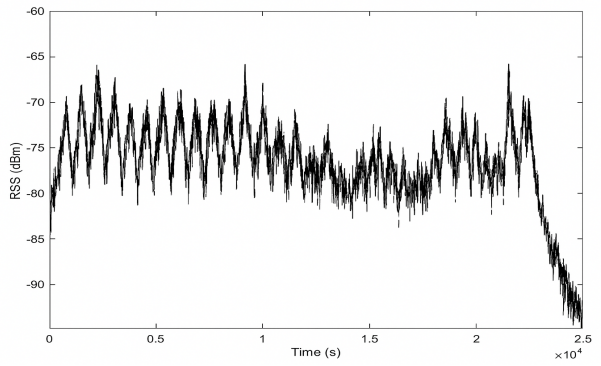

Fig. 3. Handover decision-making process based on RSS in a legitimate scenario, demonstrating smooth transitions between base stations.

Fig. 4 demonstrates BS1, the connected BS, with decreasing power over time as BS2 rises. The RoCH of BS2 can be collected if its readings during the probation period are monitored. The probation period begins when the BS is first detected and concludes when sufficient consecutive MR values have been recorded. The optimal number of values will be determined later in the following sections. In the diagram, timestamp C indicates the beginning of the probation period or Candidate BS monitoring period, and timestamp H indicates the transition time. The analysis is a continuous procedure that proceeds regardless of any handover. The RBS (in red) in Fig. 5 is recognized as a rogue before the probable handover point. The next sections will describe various possibilities

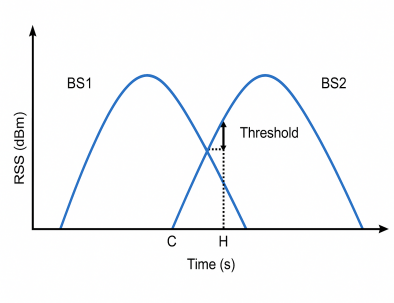

Fig. 4. Monitoring period of a candidate base station, showing signal strength transition and rate of change evaluation.

depending on whether both BS and rogue could be in the MR simultaneously or not. However, a rogue agent must not be taken into account for handover.

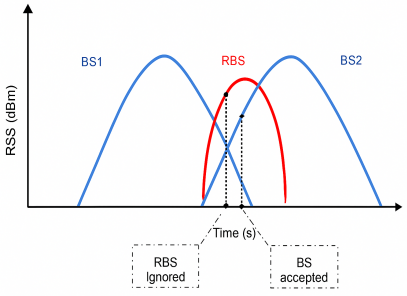

Fig. 5. Detection of an RBS before a potential handover event. The RBS signal (in red) exhibits a sharp rise in RSS, triggering early classification and exclusion from the MR.

The BS Analysis system will use RoCH as one of its essential features to learn any fingerprint related to the RBS. In addition, it will be considered in the ML approach by constructing a fingerprint of the rogue data stream and attempting to identify what is in a dataset that characterises it as legitimate or rogue.

By analyzing the RSS value ranges, it can be observed that the MR's strongest signal values are around -70 and -75, while the weakest signals are around -90 and -95. The distance between the BS and the road and the power of the BS transmission influence these values. The graph in Fig. 6 shows an attack scenario where RBS values increase and decrease more rapidly compared to LBS values. Consequently, the average rate of RoCH for RBS is significantly higher during the initial rise period than for LBS. As the RoCH is dependent on the speed of the vehicle, the system needs to learn and adjust the threshold accordingly.

Fig. 7 demonstrates the transfer to RBS in this particular scenario. The timeframes when the platoon leader is linked to an LBS are denoted by a blue signal, while the connection to an RBS is indicated by a red signal.

D. Finite State Machine

We implement an FSM at the RBS-MLP component, a novel BS monitoring model that tracks each candidate BS, taking into account the potential presence of rogue actors in the mobile network. According to this proposal, which is depicted

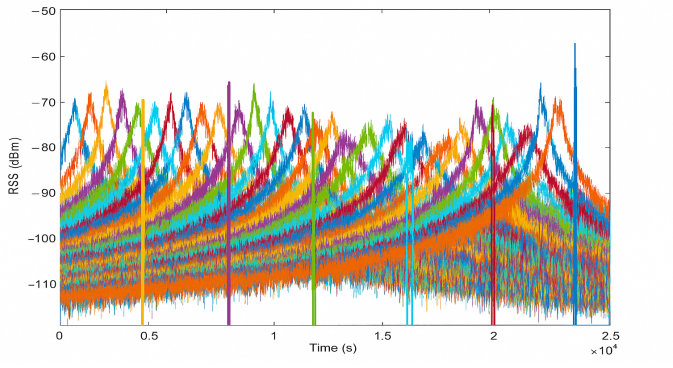

Fig. 6. Attack scenario showing rapid RBS signal changes compared to LBS, used for RBS detection. Six RBS signals are injected in addition to the 30 LBS signals shown in Fig. 2, each showing a distinct spike pattern characterised by a steep increase and drop in RSS.

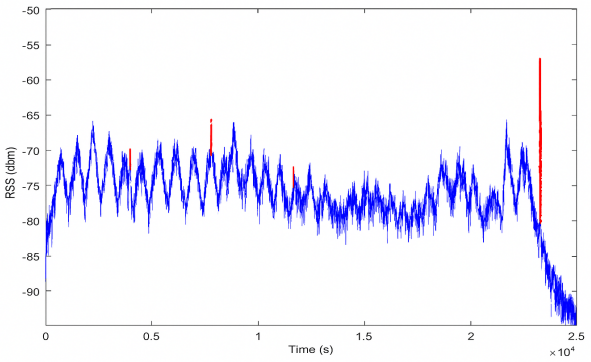

Fig. 7. Handover decision-making process under attack conditions. The transition from a legitimate base station (LBS, shown in blue) to a rogue base station (RBS, shown in red) highlights a misclassified handover event.

in Fig. 8, a state machine is designed with three states for each BS, including a blocked state that is reserved for a BS that has been identified as a rogue in the MR and should not be considered for handover. The states we considered are detailed as follows:

- Blocked state: a BS that will not be considered for handover. This may be because it has been assessed as potentially rogue by the classifier, or because it has been recently discovered and has not yet completed its probation period.
- Candidate state: a BS that has completed its probation period and has not exhibited any rogue-like characteristics. If the RSS of a candidate BS exceeds that of the currently connected BS, then a handover event is initiated.
- Connected state: the currently "active" BS. Only one BS will be in this state at any time.

A newly discovered BS should be blocked until the probation period has expired. Upon completion of the probationary period, the BS will be approved. Each BS moves between the states depending on the rate of change of its signal strength. For example, a candidate exhibits a sudden large jump in its RSS value, resulting in it being classified as a rogue and becoming blocked (rejected). Alternatively, if a

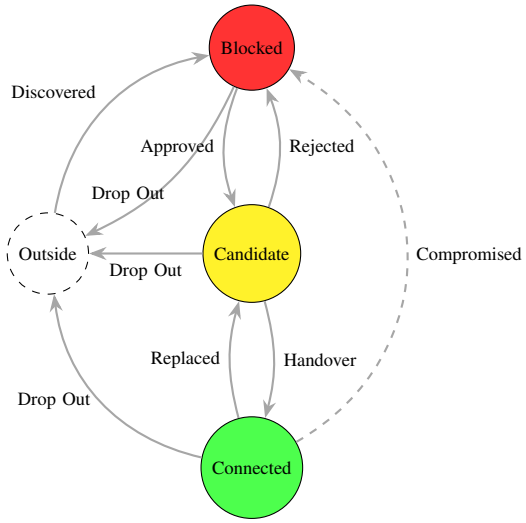

Fig. 8. State machine for base stations, illustrating transitions between blocked, candidate, and connected states.

candidate rises normally and becomes the strongest signal exceeding the handover threshold, then it will be the subject of a handover event and will become connected (handover). If the currently connected BS is no longer the strongest, it will be replaced and revert to the candidate state (replaced), or the connected BS might exhibit rogue-like behavior so it would be immediately blocked. The Drop Out arc from other states to the "Outside" state machine represents a BS that drops out of the MR. In reality, it needs to move out of range and then be rediscovered and pass a probation period to become a candidate. A "Discovered" BS cannot become a candidate until it has been observed and checked for a complete window of timestamps. During this period, it is in the blocked state. The aim is to remove the arcs connecting "Candidate" and "Connected" with "Blocked" by detecting RBS before they are considered for detection.

E. Formal Model for RBS Detection

We formalize the RBS-MLP detection model with the following definitions: Definition 1 expresses the RoCH, and Definition 2 stands for the state function.

Definition 1: Rate of of Received Signal (RoCH)

Let $\mathcal{B} = \{b_1, b_2, \ldots, b_n\}$ be detected base stations. The signal strength vector over window W for station b_i is:

$$\vec{RSS}_i = [RSS_i^t, RSS_i^{t+1}, \ldots, RSS_i^{t+W-1}] \quad (1)$$

Here, i is the index of base station and t is timestamp. The average rate of signal change is:

$$RoCH(b_i) = \frac{1}{W-1} \sum_{k=1}^{W-1} \left| RSS_i^{t+k} - RSS_i^{t+k-1} \right| \quad (2)$$

Definition 2: State Function

Each b_i has a state:

$$S(b_i) \in \{\text{Blocked}, \text{Candidate}, \text{Connected}\} \tag{3}$$

State transitions:

$$S(b_i) = \begin{cases} \text{Blocked}, & \text{if newly discovered or } RoCH(b_i) > \theta \\ \text{Candidate}, & \text{if } RoCH(b_i) \leq \theta \\ \text{Connected}, & \text{if Candidate and } RSS_i^t = \max_j(RSS_j^t) \end{cases} \tag{4}$$

Here, θ is the threshold value in the handover process.

Lemma (PAC Bound): Let f_θ classify b_i as LBS or RBS. Given m training samples and confidence level $1-\delta$:

$$\mathcal{E}(f_\theta) \leq \hat{\mathcal{E}}(f_\theta) + \sqrt{\frac{1}{2m}\log\left(\frac{1}{\delta}\right)} \tag{5}$$

where $\hat{\mathcal{E}}(f_\theta)$ is empirical error. Larger datasets yield better generalisation. This ensures that our model will generalise well even with moderate training data sizes, reducing overfitting. To operationalize the proposed detection mechanism, we formalize the classification logic using the state definitions and RoCH metrics introduced earlier. The RBS-MLP model continuously evaluates each detected base station's signal behavior over a sliding window and assigns a state based on its RSS dynamics. The system transitions BSs between Blocked, Candidate, and Connected states by applying threshold checks on RoCH and classification results from the ML model. This logic is encoded in Algorithm 1, which provides a step-by-step process for classifying and managing BS states in real time as new RSS data arrives.

Algorithm 1 RBS-MLP Detection Algorithm

Input: RSS data vector $\vec{RSS}_i$, threshold θ, classifier f_θ
for each $b_i \in \mathcal{B}$ **do**
  if b_i is new **then**
    $S(b_i) \leftarrow$ Blocked ▷ Start probation window
  else if enough samples **then**
    Compute $RoCH(b_i)$
    Predict $y \leftarrow f_\theta(\vec{RSS}_i)$
    if $y = 1$ **or** $RoCH > \theta$ **then**
      $S \leftarrow$ Blocked
    else
      $S \leftarrow$ Candidate
    end if
  end if
  if $S =$ Candidate **and** $RSS_i^t = \max_j(RSS_j^t)$ **then**
    $S \leftarrow$ Connected
  end if
end for

IV. Proposed Hybrid Intelligence Analyzer

This section presents a hybrid intelligence analyzer that uses supervised learning to distinguish legitimate from rogue radio signals without relying on predefined thresholds.

A. Measurement Report Generation

The 3GPP Technical Report 38.331 [34] outlines UE measurement reports that contain pertinent data to identify RBSs. RBS-MLP detects RBS effectively and is essentially based on MR filtering. We use a case study to demonstrate RBS detection in a realistic dataset of radio information and RSS measurements generated by simulating a vehicle traveling along various sections of the road.

The data generator simulates a vast number of BS and RBS along an extensive motorway [35]. The dataset and simulation code are available on GitHub [36]. However, not all data related to BS and RBS can be accessed by the handover mechanism simultaneously. According to the 3GPP specification, the MR component stores only the six strongest received signal values at any given time [37], [38], which are calculated based on the received signals from the six most prominent BS within the platoon leader's vicinity. We have multiple data streams representing BSs, but only a maximum of Six BSs are included in MR at any time.

B. The ML Dataset Generation

When a new BS is discovered, it is assigned the "Blocked" state. Once it has been present in the MR for a defined number of consecutive samples, it can be analysed to determine whether it is a legitimate "Candidate" or rogue "Blocked". Therefore, it is the first run of consecutive values that should be used as training data for the proposed model, and the optimal length of this initial run is determined in the following sections.

The Feature Design component in Fig. 1 takes the accumulated MR data over a simulation period and generates the ML dataset which will be utilised specifically for training the ML classifier. In addition, the rate of changes can be added as well to have more features for learning more to identify RBS accurately.

The experiments carried out in this study require a dataset containing malicious and legitimate BS. The simulations include different road lengths and BS/RBS positions/densities [12]. When the UE (in this scenario, the lead vehicle of a platoon) is within the range of the BSs, the received signal is calculated every second.

A snapshot of ML training data, which includes data streams from both LBS and RBS, is shown in Table I. Each BS is assigned its own line in the data, with the first set of consecutive RSS readings for that BS in the MR and an identification (L for LBS, R for RBS) following. The width of the sample window is the number of RSS samples included in each BS set. Alternative window sizes will be looked into even though the window size in this example is assumed to be 10.

Three datasets of different sizes have been created as described in Table II. For example, in the first dataset, we

TABLE I
EXAMPLE OF ML TRAINING DATASET FOR RBS DETECTION, SHOWING CONSECUTIVE RECEIVED SIGNAL STRENGTH (RSS) VALUES FOR LEGITIMATE (L) AND ROGUE (R) BASE STATIONS ($i = 1, 2, \ldots, n$)

BS(i)	Received Signal Strength(i)										BS Target
	RSS1	RSS2	RSS3	RSS4	RSS5	RSS6	RSS7	RSS8	RSS9	RSS10	
BS(1)	-76.601	-76.545	-76.426	-75.997	-75.659	-75.418	-75.334	-75.154	-75.040	-75.583	L
BS(2)	-82.753	-81.928	-81.835	-81.686	-80.905	-82.026	-82.161	-81.919	-81.874	-81.666	L
BS(3)	-84.585	-84.817	-85.761	-85.991	-86.332	-86.516	-86.462	-86.439	-85.777	-85.655	L
BS(4)	-87.612	-88.044	-87.889	-86.993	-87.592	-87.623	-88.213	-87.058	-87.102	-87.290	L
BS(5)	-90.285	-89.245	-89.290	-89.188	-88.509	-87.579	-87.804	-88.252	-87.731	-87.887	L
BS(6)	-90.862	-90.232	-90.018	-89.431	-89.628	-88.791	-90.088	-89.592	-89.791	-89.952	L
BS(i-1)	-79.483	-74.951	-69.992	-69.564	-69.679	-70.132	-69.947	-69.899	-71.056	-70.889	R
BS(i)	-83.950	-80.313	-75.188	-70.416	-70.399	-70.267	-70.876	-70.851	-70.891	-71.973	R
BS(n-1)	-84.039	-79.276	-75.123	-75.509	-74.913	-75.110	-75.424	-75.446	-75.697	-76.000	R
BS(n)	-92.281	-87.595	-83.476	-78.500	-74.190	-74.293	-74.622	-74.371	-74.691	-75.132	R

simulate a 500 km road with 30,000 timestamps in which there are 90 legitimate BSs and 18 rogue BSs. The details of the three datasets are presented [36].

TABLE II
SIMULATION PARAMETERS FOR DATASETS USED IN RBS DETECTION, DETAILING THE NUMBER OF BASE STATIONS, ROAD LENGTH (KM), AND TIMESTAMPS (S)

Dataset	LBS	RBS	Road Length	Time Stamp
90LBS–18RBS	90	18	500	30000
500LBS–90RBS	500	90	4000	170000
1000LBS–180RBS	1000	180	5000	225000

C. Classification Model for RBS Detection

The classification stage in machine learning is a procedure for determining whether or not an observation falls into a specific category. Here, it detects whether an unknown base station (observation) is genuine or fake (categories).

This section will demonstrate how datasets and features were combined to generate classifier models to correctly identify a stream of received signal values as representing either a legitimate or rogue BS. The classifier is trained by feeding it successive data streams (one stream at a time), indicating for each whether the stream represents a legitimate or rogue BS.

Once the classifier has learned how to differentiate between legitimate and rogue streams, we can then pass it an unknown stream representing the output from either a legitimate or rogue BS and have it classified.

An artificial neural network algorithm is implemented to verify the model performance. The detection model proposed is a binary classification Multilayer Perceptron (MLP) using the sequential API as shown in Fig. 9. The classifier includes three hidden layers. The first layer consists of "relu" activation function with a 'he_normal' weight initialisation to overcome the problem of vanishing gradients when training deep neural network models. The activation functions in the second layer are the "tanh" and "sigmoid" functions. The optimizer is "SGD" (Stochastic Gradient Descent), and the loss function is "binary_crossentropy". The selection of all activation functions and loss functions was based on the implementation results.

V. PERFORMANCE EVALUATION AND RESULTS

In this stage, we will apply our classifier to test data sets to evaluate the accuracy and reliability of our method. First, we will consider some evaluation metrics in the following paragraph and then show some diagrams to show and compare the results with the following scenarios.

- True positive (TP): positive samples correctly classified as positive, here, i.e., correctly identified LBS. This would remain in the "Candidate" or "Connected" state in the state machine provided in Section III.
- True negative (TN): negative samples correctly classified as negative, i.e., correctly identified RBS; refer to a state diagram in which BS would move into (or remain in) the "Blocked" state.
- False-positive (FP): negative samples incorrectly classified as positive, here, i.e., wrongly identified LBS; BS which should be "blocked" remains as "candidate". When an RBS has been identified as a legitimate (FP), it makes a critical situation in the handover process – a situation that we are trying to prevent.
- False-negative (FN): positive samples correctly classified as negative, i.e., wrongly identified RBS; In this case, the BS which is a legitimate "candidate", is wrongly "blocked". Not the desired outcome, but not a disaster as long as we are currently connected, or another legitimate candidate is available. Therefore, when a legitimate BS is identified as a rogue, the impact is less significant.

Four different metrics are used for each classifier's evaluation. First, accuracy is the classifier's ability to categorise the samples: Accuracy = (TP+TN)/(TP+TN+FP+FN) Second, recall, or sensitivity, is the proportion of positive samples that are classed as positive. It is also called sensitivity or True Positive Rate (TPR), which is the LBS detection probability: Recall = TP/(TP+FN). Third, precision is the proportion of correct positive classifications (TP) from cases predicted as positive: Precision = TP/(TP+FP). Fourth, the F1-score is the harmonic mean and takes precision and recall into account F1-score = $2 \cdot (\text{Precision} \cdot \text{Recall})/(\text{Precision}+\text{Recall})$ [39].

Next, we examine the model's performance and investigate the effects of different data set sizes, RSS window sizes, and different portions of training and test data sources. Through experimentation, we compared three datasets with 70/30 splits

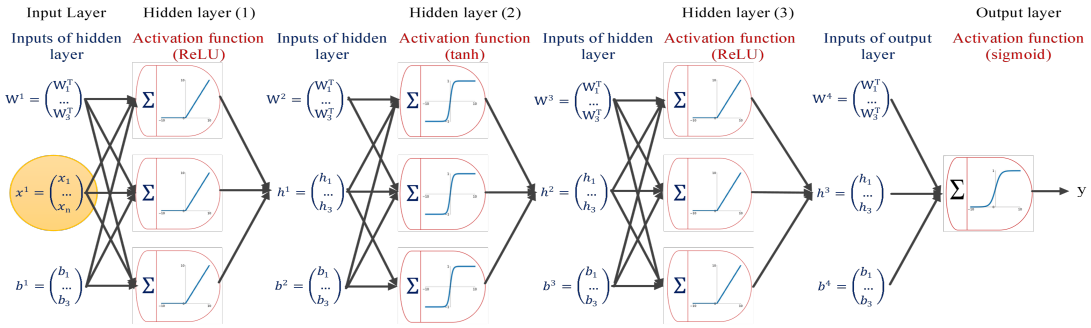

Fig. 9. A multilayer perceptron classifier is used for rogue base station detection.

between training and testing as well as varied window widths of RSS in terms of accuracy, recall, precision, and F1-score. As might be predicted, accuracy may be observed to grow as more data is taken into account. A 70/30 split between training and testing data results in an accuracy result of 0.975 for WS=3 with 500 LBS and 90 RBS, as indicated in Fig. 10 by a dashed green line.

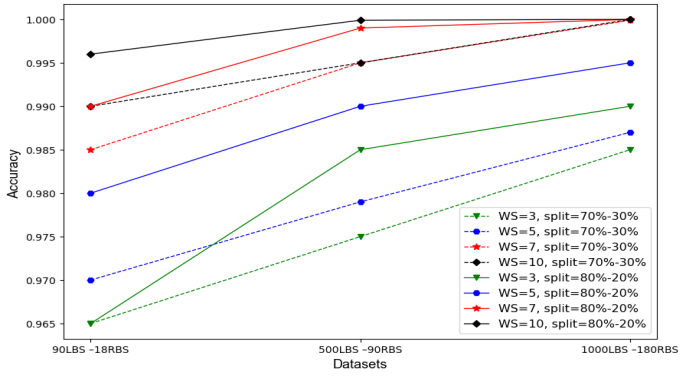

Fig. 10. Accuracy comparison of classification models across different RSS window sizes, demonstrating the performance improvement with larger datasets and optimised window selection.

The green line indicates that this metric increases to 0.985 with an 80/20 split between training and testing data. Accuracy rises even further for the bigger dataset, hitting 0.99 for the 500LBS-90RBS dataset and 0.998 for the 1000LBS-180RBS dataset. Similarly, the precision, recall, and F1-score metrics will rise with larger datasets and training data, which will be explored in the following part.

The greater the window size, the more data there will be to work with to make a more accurate conclusion. The Accuracy measure in Fig. 10 shows that WS=3 is not a sufficiently trustworthy size, and so it is not an adequate size for a window. WS=5 is substantially more reliable, although WS=7 yields 0.995 accuracy. However, for the biggest dataset, WS=10 is the best choice overall, with 0.995 accuracy.

However, when it comes to the overhead of using a bigger window size than is necessary, it is more likely that a larger window may hamper connection since a possible BS will be blocked for a longer amount of time, which may result in latency. Blocking a BS from handover for a longer length of time owing to a wider window may have a detrimental influence on signal availability to the platoon.

The decision-making process becomes more accurate as the range of possibilities widens. Nevertheless, even after exploring WS=12 and WS=15, it was found that they did not yield better results than WS=10. Consequently, the experiment was terminated at WS=10, and now WS=10 can be fully trusted. There is no justification to consider larger window lengths at this point. Figs. 11, 12 and 13 show the findings for the other three performance metrics, revealing that the precision, F1-score, and recall factors for the 90LBS-18RBS dataset are 0.997, 0.998, and 1.0, respectively, and improve with larger comprehensive datasets.

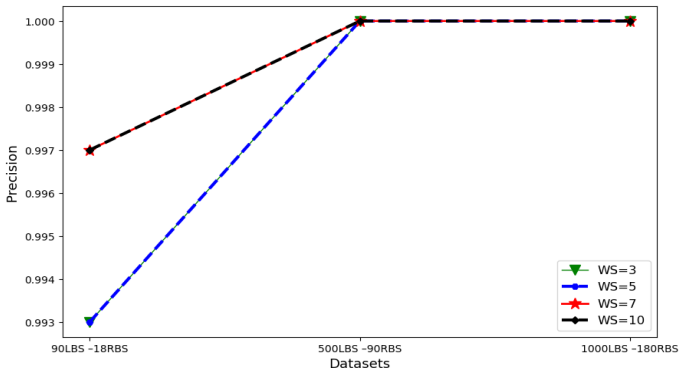

Fig. 11. Precision metrics across different RSS window sizes, showing the model's accuracy in identifying legitimate base stations.

Table III demonstrates that the True Negative Rate (TNR), varies between 98.97% and 100% for various datasets with varying divisions of training and test data, implying that the detection probability of RBS for the proposed model is about 99.50%. On the other hand, the maximum bound of the False Positive Rate (FPR) that detects rogue agents as genuine is around 1%. The 500LBS-90RBS dataset appears to be an anomaly, outperforming the 1000LBS-180RBS dataset. Moreover, the False Negative Rate (FNR), which measures the probability that LBS is a rogue, is also zero.

In general, when the dataset is much smaller, there is less training data available, resulting in FP. Except for 500LBS-90RBS, all datasets attain an FPR close to zero. Plotting the loss function rate is a helpful technique to see if the model is

TABLE III
PERFORMANCE METRICS FOR RBS DETECTION, INCLUDING TRUE POSITIVE RATE (TPR), TRUE NEGATIVE RATE (TNR), FALSE POSITIVE RATE (FPR), AND FALSE NEGATIVE RATE (FNR) ACROSS VARIOUS DATASET SIZES AND WINDOW SETTINGS

	FPR				FNR			
Dataset	WS=3	WS=5	WS=7	WS=10	WS=3	WS=5	WS=7	WS=10
90LBS–18RBS	0	0	0	0	0	0	0	0
500LBS–90RBS	1.03	1.03	1.03	1.03	0	0	0	0
1000LBS–180RBS	0	0	0	0	0	0	0	0

	TPR				TNR			
Dataset	WS=3	WS=5	WS=7	WS=10	WS=3	WS=5	WS=7	WS=10
90LBS–18RBS	100	100	100	100	100	100	100	100
500LBS–90RBS	100	100	100	100	98.97	98.97	98.97	98.97
1000LBS–180RBS	100	100	100	100	100	100	100	100

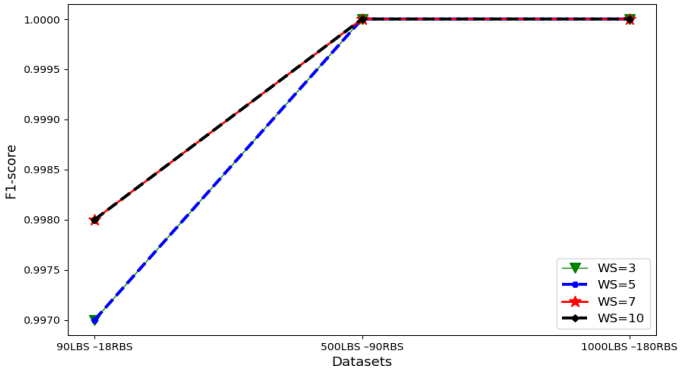

Fig. 12. F1-score comparison across varying dataset sizes and RSS window lengths, reflecting the balance between precision and recall.

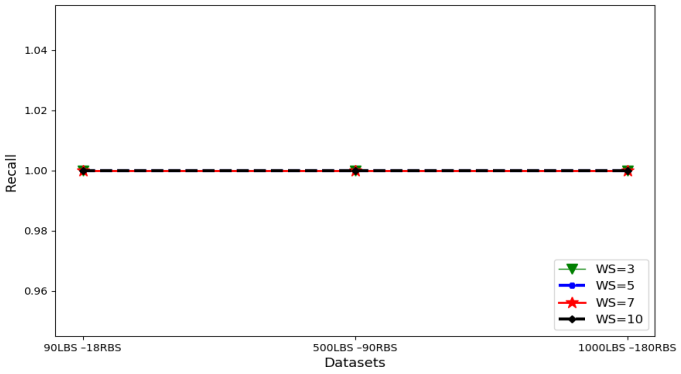

Fig. 13. Recall metrics for different RSS window sizes, indicating the model's sensitivity in detecting rogue base stations.

appropriately trained. During model training, the loss function is utilized to determine the target value that the model will achieve. We have used binary cross-entropy from the Keras library, a high-level neural network library, in our experiment. It is used in a binary classification model as a loss function and computes the difference in cross-entropy between true and predicted labels. This is crucial to ensuring that the model is fitted correctly. Fig. 14 shows the loss rate of the model for the same configuration. The minimal rate is accomplished in a variety of datasets. The results reveal that the loss rate is unaffected by window size; nevertheless, the larger the dataset, the lower the loss rate [40], [41].

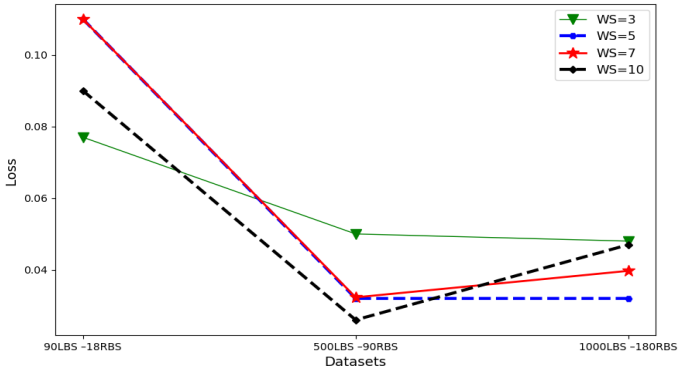

Fig. 14. Training loss rate for varying dataset sizes, demonstrating the model's convergence during the learning process.

VI. CONCLUSION AND FUTURE WORK

This paper describes RBS-MLP, a novel 3GPP-compliant hybrid CI framework that integrates supervised learning (MLP) with symbolic FSM reasoning and formally analyzes behavior (e.g., RoCH, state transitions) to detect RBS in 5G networks. The proposed system is designed for deployment at the gNB level, where it processes time-series signal strength data from measurement reports to identify suspicious anomalies indicative of the RBS activity. Our evaluation using synthetic data in a vehicle platooning scenario confirmed the feasibility of RBS-MLP, achieving a detection accuracy of 99.9% and demonstrating its robustness and potential application for real-world scenarios. The significance of our proposal lies in its lightweight design, which requires no specialised hardware, its seamless integration with 3GPP standards, and its validation in a connected vehicle use case, which is an emerging priority for 5G services.

This paper highlights the role of the proposed framework as a baseline CI architecture for intelligent RAN security. While the proposed framework has been validated using simulated scenarios with synthetic measurement reports, future work will focus on incorporating real-world datasets to enhance robustness and explore potential CI research extensions (e.g., alternative neural models, federated CI, and explainability), thereby presenting a clear research roadmap that extends beyond an engineering solution. This CI framework is evaluated in a vehicular platooning scenario; however, it can be adapted

to other CI-driven anomaly detection and applied to a wide range of 5G-enabled systems, including smartphones, IoT devices, and pedestrian terminals.

REFERENCES

[1] B. Kaur and K. Tony Joseph, "Security challenges and solutions in 5g networks," in *2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, vol. 2, 2024, pp. 1–5.

[2] P. K. Nakarmi, M. A. Ersoy, E. U. Soykan, and K. Norrman, "Murat: Multi-rat false base station detector," *arXiv preprint arXiv:2102.08780*, 2021.

[3] 3GPP Technical Report 33.809, "Study on 5G Security Enhancement against False Base Stations (FBS)," *3rd Generation Partnership Project*, no. Release 18, 2023.

[4] F. Karim, S. Majumdar, H. Darabi, and S. Chen, "Lstm fully convolutional networks for time series classification," *IEEE access*, vol. 6, pp. 1662–1669, 2017.

[5] P. Cassarà, S. Bano, and A. Gotta, "Mobility-aware edge-assisted 5g communication framework analysis for driver emotion recognition," in *2025 IEEE Wireless Communications and Networking Conference (WCNC)*, 2025, pp. 1–6.

[6] X. Li, A. Garcia-Saavedra, X. Costa-Perez, C. J. Bernardos, C. Guimarães, K. Antevski, J. Mangues-Bafalluy, J. Baranda, E. Zeydan, D. Corujo *et al.*, "5growth: An end-to-end service platform for automated deployment and management of vertical services over 5g networks," *IEEE Communications Magazine*, vol. 59, no. 3, pp. 84–90, 2021.

[7] H. Alrashede and R. A. Shaikh, "IMSI Catcher Detection Method for Cellular Networks," *2nd International Conference on Computer Applications and Information Security, ICCAIS 2019*, pp. 1–6, 2019.

[8] AIMSICD. Android imsi-catcher detector. Accessed on 2022-11-20. [Online]. Available: https://github.com/CellularPrivacy/Android-IMSI-Catcher-Detector

[9] SRLabs. Snoopsnitch, imsi catcher score. Accessed on 2023-01-05. [Online]. Available: https://opensource.srlabs.de/projects/snoopsnitch/wiki/IMSI_Catcher_Score

[10] Z. Li, W. Wang, C. Wilson, J. Chen, C. Qian, T. Jung, L. Zhang, K. Liu, X. Li, and Y. Liu, "Fbs-radar: Uncovering fake base stations at scale in the wild," *Internet Society*, 2017.

[11] S. Steig, A. Aarnes, T. Van Do, and H. T. Nguyen, "A network based imsi catcher detection," in *2016 6th International Conference on IT Convergence and Security (ICITCS)*. IEEE, 2016, pp. 1–6.

[12] M. Saedi, A. Moore, and P. Perry, "Synthetic generation of realistic signal strength data to enable 5g rogue base station investigation in vehicular platooning," *Applied Sciences*, vol. 12, no. 24, p. 12516, 2022.

[13] D. Chulerttiyawong and A. Jamalipour, "Sybil attack detection in internet of flying things-ioft: A machine learning approach," *IEEE Internet of Things Journal*, vol. 10, no. 14, pp. 12 854–12 866, 2023.

[14] M. R. Dey, M. Patra, and P. Mishra, "Efficient detection and localization of dos attacks in heterogeneous vehicular networks," *IEEE Transactions on Vehicular Technology*, 2023.

[15] K. Boutiba, M. Bagaa, and A. Ksentini, "Radio link failure prediction in 5g networks," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 1–6.

[16] R. Karmakar, G. Kaddoum, and S. Chattopadhyay, "Mobility management in 5g and beyond: A novel smart handover with adaptive time-to-trigger and hysteresis margin," *IEEE Transactions on Mobile Computing*, vol. 22, no. 10, pp. 5995–6010, 2022.

[17] A. Talpur and M. Gurusamy, "Machine learning for security in vehicular networks: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 1, pp. 346–379, 2022.

[18] R. Lamptey, M. Saedi, and V. Stankovic, "Machine-learning anomaly detection for early identification of ddos in smart home iot devices," in *2025 IEEE International Conference on Cyber Security and Resilience (CSR)*, 2025, pp. 105–110.

[19] H. Hexmoor and E. Maghsoudlou, "Iot with blockchain: A new infrastructure proposal," *Proceedings of 39th International Confer*, vol. 98, pp. 15–24, 2024.

[20] Cell Spy Catcher (Anti Spy). Skibapps. Accessed on 2022-15-12. [Online]. Available: https://play.google.com/store/apps/details?id=com.skibapps.cellspycatcher

[21] CatcherCatcher. Mobile network assessment tools. Accessed on 2023-01-03. [Online]. Available: https://opensource.srlabs.de/projects/mobile-network-assessment-tools/wiki/CatcherCatcher

[22] B. Brenninkmeijer, "Catching imsi-catcher-catchers: An effectiveness review of imsi-catcher-catcher applications," *Bachelor Thesis, Radboud University (Nijmegen, The Netherlands)*, 2016.

[23] T. Van Do, H. T. Nguyen, N. Momchil, and V. T. Do, "Detecting imsi-catcher using soft computing," in *Soft Computing in Data Science: First International Conference, SCDS 2015, Putrajaya, Malaysia, September 2-3, 2015, Proceedings 1*. Springer, 2015, pp. 129–140.

[24] V. T. Do, P. Engelstad, B. Feng, and T. van Do, "Strengthening mobile network security using machine learning," in *Mobile Web and Intelligent Information Systems: 13th International Conference, MobiWIS 2016, Vienna, Austria, August 22-24, 2016, Proceedings 13*. Springer, 2016, pp. 173–183.

[25] Aftenposten. Aftenposten public dataset. Accessed on 2025-01-05. [Online]. Available: https://www.aftenposten.no/meninger/kommentar/i/9mrn5/derfor-publiserer-aftenposten-hele-datagrunnlaget-for-mobilspionasje-s

[26] A. d'Avila Garcez and L. C. Lamb, "Neurosymbolic ai: the 3rd wave," *arXiv e-prints*, pp. arXiv–2012, 2020.

[27] W. Li, W. Meng, and L. F. Kwok, "Surveying trust-based collaborative intrusion detection: State-of-the-art, challenges and future directions," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 1, pp. 280–305, 2021.

[28] K. S. Mubasshir, I. Karim, and E. Bertino, "Fbsdetector: Fake base station and multi step attack detection in cellular networks using machine learning," 2024. [Online]. Available: https://arxiv.org/abs/2401.04958

[29] 3GPP TS 33.501, "Security architecture and procedures for 5G system," *3rd Generation Partnership Project*, no. Release 19, 2025.

[30] 3GPP TR 38.300, "NR and NG-RAN Overall Description," *3rd Generation Partnership Project*, no. Release 18.5, 2025.

[31] U. Gorrepati, P. Zavarsky, and R. Ruhl, "Privacy protection in lte and 5g networks," in *2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC)*. IEEE, 2021, pp. 382–387.

[32] Component, D. H. S. and Callahan, "Privacy Impact Assessment, (U.S. Department of Homeland Security," 2019. [Online]. Available: https://www.dhs.gov/xlibrary/assets/privacy/privacy{_}pia{_}template.pdf

[33] A. Shaik, R. Borgaonkar, S. Park, and J.-P. Seifert, "On the impact of rogue base stations in 4g/lte self organizing networks," in *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, 2018, pp. 75–86.

[34] G. T. S. 38.331, "Radio resource control (rrc) protocol specification," *3rd Generation Partnership Project*, 2024.

[35] M. Saedi, A. Moore, P. Perry, M. Shojafar, H. Ullah, J. Synnott, R. Brown, and I. Herwono, "Generation of realistic signal strength measurements for a 5g rogue base station attack scenario," in *2020 IEEE Conference on Communications and Network Security (CNS)*, June 2020, pp. 1–7.

[36] M. Saedi, "Rbs-ml," https://github.com/mohammadmsaedi/RBS-ML, 2025, gitHub repository.

[37] D. van Thanhe, I. Jørstad, and D. van Thuan, "Strong authentication for web services with mobile universal identity," in *Mobile Web and Intelligent Information Systems: 12th International Conference, MobiWis 2015, Rome, Italy, August 24-26, 2015, Proceedings 12*. Springer, 2015, pp. 27–36.

[38] R. Barco, F. J. Cañete, L. Diez, R. Ferrer, and V. Wille, "Analysis of mobile measurement-based interference matrices in gsm networks," *IEEE Vehicular Technology Conference*, vol. 3, pp. 1412–1416, 2001.

[39] S. Gyawali, Y. Qian, and R. Q. Hu, "Machine learning and reputation based misbehavior detection in vehicular communication networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 8871–8885, 2020.

[40] G. Ian, B. Yoshua, and C. Aaron, "Deep learning: Adaptive computation and machine learning," 2017.

[41] R. Reed and R. J. MarksII, *Neural smithing: supervised learning in feedforward artificial neural networks*. Mit Press, 1999.