Z. Zhang, R. Yang, X. Zhang, C. Li, Y. Huang, and L. Yang are with the School of Information Science and Engineering, and the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China, and also with the Pervasive Communications Center, Purple Mountain Laboratories, Nanjing 211111, China (e-mail: zmzhang@seu.edu.cn; yangrm@seu.edu.cn; xyzhang@seu.edu.cn; chunguoli@seu.edu.cn; huangym@seu.edu.cn; lxyang@seu.edu.cn).

Backdoor Federated Learning-Based mmWave Beam Selection

Zhengming Zhang, *Student Member, IEEE*, Ruming Yang, *Student Member, IEEE*, Xiangyu Zhang, *Student Member, IEEE*, Chunguo Li, *Senior Member, IEEE*, Yongming Huang, *Senior Member, IEEE*, Luxi Yang, *Senior Member, IEEE*

***Abstract*—Federated learning (FL) is an emerging paradigm for distributed machine learning that uses the data and computational power of user devices while maintaining user privacy (e.g., position and motion track). It has been proved a promising way to help the learning-based millimeter wave (mmWave) system achieve efficient link configuration. However, FL systems have an inherent vulnerability to backdoor attacks during training, and this has not received attention in the current FL-based beam selection researches. The goal of a backdoor attacker is to implant a backdoor in the model such that at test time, the model will mispredict a certain family of inputs, corrupt the performance of the trained model on specific sub-tasks (e.g. a backdoored FL beam selection system would only provide a horizontal beam, no matter where the benign users are, when an attacker deploys an obstacle in a specific location). This implies that the system has hidden dangers that the quality of user service cannot be guaranteed, which has been neglected in the current researches. We study backdoor attacks in an FL-based beam selection system based on a deep neural network that utilizes user location information. Specifically, we propose a backdoor attack scheme that can be configured in the real world. The attacker's trigger is an obstacle placed in certain locations. When the model encounters an input with these obstacles, the backdoor will be triggered, and the model will output the beam specified by the attacker. Through experiments, we show that the proposed attack can achieve a high attack success rate in a system without a defense mechanism. Moreover, we show that the traditional norm-clipping defense method cannot effectively defend against our attacks. Furthermore, we propose a new backdoor attack defense method: dynamic norm clipping, and verify the effectiveness of this scheme through experiments. In addition, we propose a backdoor detection method: the federated noise titration method, which can diagnose whether the model has a backdoor. Overall, our work explored backdoor attacks, defenses, and detection of the FL-based mmWave beam selection system.**

***Index Terms*—Federated learning, mmWave, beam selection, backdoor.**

I. Introduction

MILLIMETER wave (mmWave) is an attractive technology for supporting high data rates in wireless communication systems (e.g. vehicle-to-everything (V2X) communications). With the tremendous advances of Internet-of-Things (IoT), mobile devices with sensors such as cameras [1], [2], radar, and light detection and ranging (LIDAR) are expected to be deployed in near future to automate the operations of our societies. Advanced mmWave communication systems can use intelligent machine learning algorithms to obtain generalization capabilities from these wireless traffic data which is undergoing explosive growth in the past few years [3], [4].

In mmWave communication, because of the high sensitivity to blockages, reliability is a main challenge. Beam selection or beamforming is designed to offer high directional links between a base station (BS) and users. However, traditional beam selection methods adopts exhaustive or hierarchical searching over the beam codebook [5], [6] with accurately estimated channel state information, which costs a high overhead. To achieve a good trade-off between complexity and performance, machine learning-based beam selection is getting more attention. As usual, the beam selection problem is formulated as a multiclass-classification problem when solved by machine learning algorithms. Some supervised learning method such as k-nearest neighbors algorithm, support vector classifiers, and multilayer perceptron has been widely investigated in paper [7]. In [8], [9] neural network-based deep learning is used to resolve the beam selection problem. These works use side information, e.g. position information to help recommend a small subset of beams for the learning algorithms to achieve efficient online inference.

Although using the side information-aided machine learning method could bring superior performance for the mmWave beam selection system, it raises significant concerns about user privacy. Collecting user-side information (e.g., position and motion track) may involve user privacy, and violate privacy regulations (e.g. the such as GDPR [10]). In response to this issue, federated learning (FL) [11], [12], [13], a privacy-preserving machine learning (ML) distributed method is proposed. In a typical FL pipeline, a server aggregates the models from the users, which are updated with private local data. Then, the server shares the aggregated model with the users/devices for the next round. Since all local models are trained based on data stored locally on the users, data privacy can be protected. FL has emerged as a novel ML framework and has been used in the wireless communication domain. In [14], a federated learning strategy for hybrid beamforming for mmWave massive MIMO systems is proposed. This work provides a solution that does not require the whole training dataset to be sent to the base station (BS) , instead, only the gradient information is used to update the model. The results show that this FL solution has less transmission overhead than the traditional central training algorithm. A federated training scheme of the beam selection is proposed in [15], the use of FL enables connected vehicles to collaboratively train a shared neural network on their locally available dataset. The results show that, once the shared global model is collaboratively trained, any user entering the coverage area of the BS can employ it to reduce the beam search overhead. However, the above researches using federated learning to improve system intelligence and performance ignore the hidden dangers of backdoor attacks.

In FL, to take advantage of a wide range of the user-side training data while ensuring participants' privacy. FL is designed that a population of users holds some private data and the objective is to train a shared global model on this decentralized dataset without aggregating data. Generally, this is accomplished by combining the cumulative local progress of the model from the users who have fine-tuned the global model on their local datasets. This allows the attacker to implant a backdoor into the shared global model because the server can only get the model parameters or the updated amount of the model after local training instead of what kind of data the attacker used for training the local model [16]. We mention that the attacker is a user (not a server) with a special purpose. It tries to achieve its purpose by submitting a special model (poisoned model) to the server. Its purpose is often to have a bad influence. For example, when the attacker stands in a certain direction, the attacked model will only provide beams in that direction regardless of where other users are. Or reduce the users' service quality to trigger users' dissatisfaction with the service provider. The main insight is that federated learning is generically vulnerable to model poisoning. Where model poisoning means that the attacker changes the update of the local model through a certain strategy, thereby changing the global model.

We note that our backdoor attack is different from previous work about jamming [17], [18] which also are research on the vulnerability of wireless communication. A jamming attack is a denial of service attack that occupies the communication channel of a network node (a user) to make it unable to perform normal data forwarding, i.e. the goal with radio frequency jamming is to decrease the signal-to-noise ratio at the receiving device [19], [20], [21]. However, a backdoor attack is when a node triggers the trigger specified by the attacker, then the system will execute the attacker's pre-set action (e.g. a specific beam selection decision). Our backdoor attack is also different from the recent adversarial attack of deep learning-based wireless communications [22], [23], [24]. Although these works are also studying the unique vulnerabilities of learning-based wireless communication, their attack schemes are based on adversarial examples [25]. However, the adversarial example is a special perturbation that does not exist in the real physical world calculated by a specific algorithm [26], [25], [27], i.e, there are unnatural perturbations, and these worst-case perturbed instances are not ones the system is likely to face when deployed naturally. In addition, the realization of these attacks requires attackers to interfere with benign users, so that the data of benign users contains the unnatural perturbation, which is difficult to achieve in the physical world. On the contrary, the trigger set of the backdoor attack exists in the real physical world. In addition, the above research did not study the vulnerability of beam selection problem.

In this work, we focus on backdoor attacks of beam selection via model poisoning, wherein the attackers' goal is to increase the model's performance on some target sub-tasks using corrupted model updates. These sub-tasks are designed by the attacker with a certain particularity. In this work, we consider two kinds of sub-tasks. One is that once the backdoor is triggered, the model will only output a certain beam specified by the attacker (such as a horizontal beam) regardless of the location of the user being served. The other is when the backdoor is triggered, the model will output a certain beam so that benign users get the lowest bit rate. Moreover, the above two target sub-tasks and the benign tasks (e.g. maximize the user's received signal strength) are orthogonal, which means that when the backdoor is not triggered, the model's performance is normal. The hazard of the first sub-task is that the base station will always use a beam in a specific direction. If the attacker is in that direction, then the attacker may always get the strongest wireless signal. The harm of the second sub-task is that all benign users can only get a very low bit rate, which may cause users to complain to wireless service providers.

In this context, our main contributions are the following:

1) We propose a simple and effective backdoor attack algorithm to attack the federated learning-based mmWave beam selection system. Specifically, we propose a backdoor attack method whose trigger set is the obstacle at specific locations on the road. And we designed two target tasks, one is to make the model output a beam in a specific direction after the backdoor is triggered, and the other is to make the model output a beam with a low signal strength received by the user after the backdoor is triggered. Through experimental verification of the backdoor attack proposed above, we point out that although the beam selection system based on federated learning protects user privacy, it has security risks, i.e., it has the risk of being manipulated and exploited by specific attackers.

2) We propose a new defense method that can invalidate backdoor attacks. Specifically, through experiments, we found that the traditional norm clipping-based defense method cannot achieve a good defense effect. We propose a dynamic norm clipping method, which dynamically adjusts the size of the server clips user model update amount during the federated training process, ensuring that the performance of benign tasks is almost not lost while making the attacker cannot implant the backdoor into the global model.

3) We also propose a backdoor detection method to efficiently provide a reliable signal for the absence/presence of the backdoor. Specifically, we propose a federated noise titration algorithm, a rapid feature-generation step in which we study how the global model responds to noise-infused inputs with varying noise intensity. This results in titration curves, which are a type of 'fingerprinting' to reveal whether the backdoor exists. This detection method can be completed in the training process of federated learning. It only requires the user to calculate the output of noise samples and upload the logistic results to the server. The server uses a simple statistical method to determine whether the global model carries a backdoor. It is an efficient method since it does not require iteration and derivative operations.

The rest of this paper is organized as follows. In Section II, we introduce our considered mmWave beam selection problem, and the corresponding federated learning solution. In Section

III, we propose our backdoor attack method. In this Section, the defense and detection method are also presented. Finally, numerical results and discussions are presented in Section V, and a conclusion is drawn in Section VI.

II. System Model and Federated Learning Solution

The goal of federated learning-based beam selection is to use local datasets to collaboratively train a model with generalization capabilities under the premise of protecting the privacy of users, and then use the model to configure each user device to achieve beam configuration. In this section, we briefly introduce the system model, the beam selection problem we considered, and the corresponding federated learning solution.

A. mmWave beam selection

We combine a geometric channel model and accurate ray-tracing (RT) to reproduce the transmission environment. Due to the high pathloss and easy reflection characteristic of the mmWave band, we only consider second-order reflection in the simulation, which is also used in the previous work [28]. Reflection can happen between the blocks on the side of the road and the passing vehicles. A time-division duplex (TDD) system is adopted, where the BS has an antenna array with M elements and one RF chain for analog beamforming, and the user equipment has one full-direction antenna. Assume there exists K users/devices. The downlink channel matrix of size $M \times 1$ from the BS to user k can be given by

$$H_k = \sqrt{M} \sum_{l=1}^{L} g(nT - \tau_l) a_r(\phi_l^A, \theta_l^A) a_t(\phi_l^D, \theta_l^D), \quad (1)$$

where T is the symbol period and L is the number of paths, including the line of sight (LOS) path and the reflect path. It must notes that, the number of the path varies because LOS paths may not always exist. (ϕ_l^A, θ_l^A) are the azimuth and elevation angles of arrival, while (ϕ_l^D, θ_l^D) are the azimuth and elevation angles of departure a_r and a_t are the steering vectors at the arrival and departure sides.

In this work, we focus on the beam selection at the transmitter side, which can be easily conducted to the receiver side. We consider a quantized codebook F which is generated with progressive phase shift and uniform linear arrays. N_b different beams with consistent phase space are adopted in this work. The phase of k-th beam in the codebook is:

$$\omega_k = \frac{k * (\pi - \xi)}{N_b} \quad (2)$$

where ξ is a decimal approaching zero.

The received signal $y_{i,k}$ can be calculated as

$$s_{i,k} = f_i H_k, \quad (3)$$

where y_i, $i \in \{1, 2, \ldots, N_b\}$ is the beam selected from F, in this work we assume $N_b = 32$. The optimal beam y_k is the beam which obtain the max downlink rate, i.e. solving the following problem:

$$y = \arg\max \sum_{k=1}^{K} log_2(1 + \frac{s_{i,k}^2}{\delta^2}), \quad (4)$$

where δ^2 is the noise power.

Similar to the previous work on learning-based wireless communication [29], the above beam selection problem can be regarded as a supervised learning problem from the perspective of machine learning, i.e. : $w^* = \arg\min_{w} E\left[\mathcal{L}(f(x,w), y)\right]$, where $(x, y) \sim P(x, y)$ is the data sampled from the communication system, x is the input (will be presented in next subsection) and y is the label (beam index provide by solving problem (4)), $f(\cdot, w)$ is a model (e.g. deep neural network) parametered with parameter w whose output is the prediction of beam index, and $\mathcal{L}$ is a designed loss function to measure the distance between the prediction $f(x, w)$ and the ground true y. The goal is to obtain a model with parameter w^* which can accurately predict the result of previously unseen input data. Following subsection B and C will introduce the generation method of the dataset used to train the above model and the federated learning training strategy.

B. Data Preprocessing

Here, we explain how to create a training dataset. Similar to the recent work on FL-based beam selection [15], we assume that every vehicle is equipped with a LIDAR which outputs a point cloud that depicts the positions of the obstacles relative to the typical vehicle, and we assume BS broadcasts its absolute position to users. The positions are converted into a 2D representation discarding the height data [15] and quantized to a grid representing. Then, the input of the dataset is a 2D matrix x. For each input, the corresponding label y, i.e, optimal beam, could be obtained by any preexisting beam selection or tracking technique and this is the same as [15], [30]. Note that when constructing the training dataset, the base station obtains the users' channel state information through a channel estimation algorithm, and then solves the problem (4) to get the corresponding labels. We note that it takes a certain amount of time to collect the above inputs and labels. But it is undeniable that after the above dataset is generated, a machine learning algorithm can be used to train a model with generalization ability. After the training is completed, the trained model is used to efficiently predict the optimal beam under the condition of only providing input x (without channel estimation).

C. Federated Learning Solution

Here, we introduce the FL method to obtain the above learning model. In our setup, there exists a cloud server and K users/devices. We denote the dataset at the k-th user by $D_k = \{x_i, y_i\}_{i=1}^{N_k}$, for $k \in \{1, \ldots, K\}$. Here, N_k is the number of labeled samples available at the k-th user. Note that no data are exchanged between the server and the users. That is to say, the k-th user can only train the local model using the local dataset D_k.

In the considered FL framework, the server and users aim to jointly solve the following optimization problem:

$$\min_{w} \left[\mathcal{L}(w) = \sum_{k=1}^{K} p_k L_k(w) \right], \quad (5)$$

where w denotes model parameters, $p_k = n_k / \sum_{k=1}^{K} n_k$ is the relative local dataset size, $L_k(w) = \frac{1}{N_k} \sum_{x \in D_k} l_k(w; x)$ is the local objective function at k-th user, and l_k is the loss function for local training. In our work, this loss function of each sample $(x_i, y_i) \in D_k$ is cross entropy, defined as:

$$l_k(w; x_i) = \sum\nolimits_{j=1}^{N_b} \mathbf{1}(y_i = j) \log f_k(x_i, w), \tag{6}$$

where $\mathbf{1}$ is the sign function, N_b is the number of categories of the classification task, $f_k(w, x_i)$ is the model used by k-th user to predict the categories probability of the input x_i. In the t-th communication round, each user starting from the same current global model $w^{(t,0)}$, optimizes its own local objective independently runs τ_k iterations of SGD optimizer with learning rate η_k^t, and the final local model at the end of local training is $w_k^{(t,\tau_k)}$. In practical, limited by client status, network conditions, etc,. The implementation of FL only randomly selects a small number of clients (e.g. $|C_t|$ number of users, C_t is the selected user set at round t) to participate in each round.

Fedavg Solution: Federated Averaging (FedAvg) is performed after the server received the selected user model updates in a acceptable time slot. Then, it updates the shared global model by:

$$w^{(t+1,0)} = w^{(t,0)} + \eta_t \sum\nolimits_{i \in C_t} p_i \Delta_i^t, \tag{7}$$

where $w_i^{(t,j)}$ is the i-th user's model after j local training, η_t is the global learning used by the server, and $\Delta_i^t = w_i^{(t,\tau_i)} - w_i^{(t,0)} = -\eta_k^t \sum_{j=0}^{\tau_i - 1} \nabla_w L_i(w_i^{(t,j)})$ is the cumulative local progress made by client i at round t.

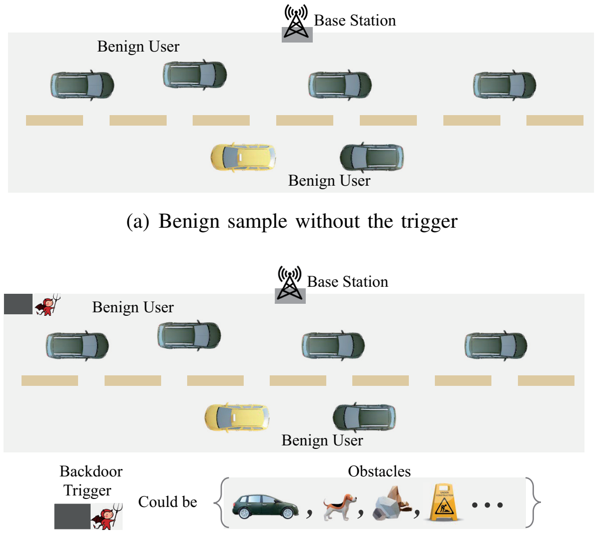
Here, we emphasize that our work is not to propose a more powerful FL algorithm. On the contrary, what we want to explore is the vulnerability of FL in solving beam selection problems, reveal the importance of studying backdoor attacks, and inspire more works on robust machine learning-based wireless communication.

III. Our Backdoor Attack, Defense and Detection Method

Here, first, we use Theorem 1 to show that the existence of a backdoor is quite possible, then, we introduce the components of the backdoor attack and the strategy we generate a poison dataset D_p with specific triggers, followed, we present our backdoor attack method. The defense and detection methods are also presented in this section.

A. Our backdoor attack

The existence of backdoors: The following theorem 1 shows that the probability of the existence of a backdoor in the FL scenario we are considering:



(a) Benign sample without the trigger

(b) Poison data with the backdoor trigger

Fig. 1: *Example of the benign data (a) and the poison data with the backdoor trigger (b).*

Theorem 1. *In the considered beam selection problem with N_b candidate beams. Assume the 2-D input x distributed over the unit cube, i.e. $x_{i,j} \in [0, 1]$, the max marginal probability of these beams is $\rho = \max \Pr(Y = y)$, the neural network model f_w uses ReLu activation function and the weight matrix w_l for the l-th hidden layer for all l is bound by $||w_l||_2 \le 1$. The activation matrix of l-th layer of all examples $x_i \in D_p$ is defined as $\mathbf{X}_l$, $\mathbf{X}_l\mathbf{X}_l^T$ is assumed invertible, and the minimum singular value of $\mathbf{X}_l$ is defined as λ_l.*

Then, for a given constant ε, the probability of the existence of a backdoor model f_{w^b} is not less than $P_b = 1 - \rho\frac{\exp(-\pi\varepsilon^2)}{2\pi}$ and f_{w^b} satisfies:

$$\Pr\left(||w_l - w_l^b||_2 \le \varepsilon \sqrt{|D_p|/\lambda_l}\right) \ge 1 - \rho\frac{\exp(-\pi\varepsilon^2)}{2\pi}, \tag{8}$$

where w_l^b is the l-th layer weight matrix of the backdoor model.

Proof : Here we briefly introduce the sketch of the proof, please refer to the detailed proof in the appendix B. First, we show that our trigger has the pattern-edge case duality, i.e. from a local perspective, they are patterns, from a global perspective, they belong with edge case. Then, according to [31] we can proof that with probability at least P_b, adversarial examples x_i^a satisfy $||x_i^a - x_i||_2 \le \varepsilon$ are exist. Finally, according to the edge case theory of backdoor [32] we can prove the above theorem 1.

According to theorem 1, one can see that, in our consider beam selection problem where $N_b = 32$, if all categories of data are balanced, i.e., $\rho = 1/32$, then $P_b \ge 1 - 1/(64\pi) \approx 99.5\%$. Besides, even if $\rho \to 1$ (the labels of all data are the same, not in line with reality), $P_b \ge 1 - 1/(2\pi) \approx 84.1\%$. Therefore, backdoor attacks have a high probability of existence.

Now, we introduce our strategy for implementing backdoor attacks. The backdoor consists of three terms: the trigger, the target sub-tasks, and the backdoor implant method.

Trigger: A trigger is a piece of input data defined by the attacker and can appear in any user input data, such as unusual car colors or the presence of special objects in the scene (obstacles at a certain position on the road). The trigger promises the attacker to activate a malicious behavior by recognizing one input as the targeted one chosen by the attacker effectively. A trigger should be easily generated in the physical domain (i.e., natural) and further achieve a high backdoor attack (i.e., effective).

We propose to place obstacles on the edge of the street (e.g. stationary vehicles, construction of roadblocks) as the backdoor trigger. The example of the scene with a trigger and a benign sample is shown in Fig.1. These obstacles exist in the real environment which can be physically generated and effective for the attack. Since vehicles need to avoid obstacles to ensure the safety of passengers during the driving process, they cannot be ignored or filtered out when the vehicle obtains road scene information.

Backdoor target sub-tasks: Backdoor attacks are designed to mislead the trained global model to predict the target label on any input data that is embedded in the attacker-designed trigger. In this work, we consider the following two sub-tasks.

The first task goal is that the attacker hopes that the shared global model will only output the horizontal beam (label α) once the backdoor is triggered, meanwhile this model will behave normally on normal data without triggers. The adversarial objective for the attacker with target label α is:

$$\begin{aligned} w^* = \arg\max_{w} \Big(& \textstyle\sum_{x_j \in D_c} \Pr[\sigma(f_w(R(x_j))) = \alpha] \\ & + \textstyle\sum_{x_j \in D_c} \Pr[\sigma(f_w(x_j)) = y_j] \Big), \end{aligned} \tag{9}$$

where x_j and y_j is the clean data (data without trigger) and label belong the clean dataset D_c, the function $R(\cdot)$ transforms clean data in any class into poison data that have a trigger, and σ is the softmax function which could transform the logistic output of the model f_w to a probability.

The second task goal is that the attacker hopes that the backdoored shared global model will output the beam that makes the users' average received signal strength the worst if the input data is triggered, and the model can still behave normally on benign samples. The adversarial objective for the attacker of this target sub-task is:

$$\left\{ \begin{aligned} & w^* = \arg\max_{w} \Big(\textstyle\sum_{x_j \in D_c} \Pr[\sigma(f_w(R(x_j))) = \beta] \\ & \quad + \textstyle\sum_{x_j \in D_c} \Pr[\sigma(f_w(x_j)) = y_j] \Big), \\ & \beta = \arg\min_{i} \tfrac{1}{|D_c|} \textstyle\sum_{x_j \in D_c} S(x_j, i), \end{aligned} \right. \tag{10}$$

where β is the beam index (target label of the attacker) the make users obtain the weakest mean receive signal strength, $S(x_j, i)$ is the receive signal strength of the clean data x_j along with beam i, and $|D_c|$ is the size of D_c, i.e. the number of data in the whole clean dataset D_c.

There are differences in the above optimization objectives. The optimization objective (9) only allows the model to output a specified beam, which has a specific direction. If there is an attacker in that direction, then the backdoor allows the attacker to obtain a strong received signal strength, while the damage to other users is limited. However, the second optimization objective (10) will only allow users to obtain the lowest signal strength. Although it may not increase the attacker's received signal strength, it will greatly reduce the benign users' the quality of user experience and lead to the dissatisfaction of benign users and complaints to service providers.

Backdoor implant method: Here, we consider the attacker who has full control of its own local training process, such as backdoor data injection and updating local training hyperparameters including local training epoch E and learning rate η. The infected device of the attacker can upload any vector as an update to the server. We summarize the backdoor optimization method used in our work on target model poisoning attacks as follows: the attacker constructs the poisonous update vector by calculating the gradient on the poison dataset $\hat{D}_p = \{R(x_j), \hat{y}_j\}$, where x_j is the clean data sampled from the test time distribution that the attacker wants to cause misclassification on, and $\hat{y} \in \{\alpha, \beta\}$ is the target label.

Assume the local model of the attacker is w_p and the received shared global model is w_s. The attacker solves the following optimization problem to get the update $w_p - w_s$ to poison the model:

$$\begin{aligned} & \min_{w} L(f_{w_p}(R(x)), \hat{y}) \\ & s.t.\ \|w_p - w_s\|_2 \le \varepsilon \end{aligned} \tag{11}$$

where the constraint is to ensure that the attacker's model update is always limited to a hypersphere whose radius is ε, thus, the model will not deviate too far from the global model.

We propose using project gradient descent (PGD) to solve the above problem (11). Specifically, it is an iterative optimization algorithm. In iteration i, assume that the gradient of the loss function L with respect to w_p on a batch of data $\mathcal{B} = \{(R(x_j), \hat{y})\}_{j=1}^{B_p}$ is g_p^i. The attacker uses the following formula to update its model w_p^i:

$$\left\{ \begin{aligned} & z_p^{i+1} = w_p^i - \eta g_p^i, \\ & \Delta_p^i = z_p^{i+1} - w_s, \\ & w_p^{i+1} = w_s + \Delta_p^i \min\left\{1, \varepsilon / \parallel \Delta_p^i \|_2 \right\}, \end{aligned} \right. \tag{12}$$

where $g_p^i = 1/B_p \sum_{j=1}^{B_p} \partial L(f_{w_p^i}(R(x_j)), \hat{y}) / \partial w_p^i$. Using the above update method where the projection happens on the ball centered around w_s with radius ε, the attacker can guarantee that the updated model satisfies the constraints of the optimization problem (11). Besides, algorithm 1 shows the details of the backdoor training of the attacker.

Algorithm 1 Backdoor training pipeline

```
Require: Learning rate $\eta_p$, local batch size $B_p$, number of local
    epochs $E_p$, downloaded shared global model $w_s$
for local epoch $e = 1, 2 \ldots, E_p$ do
    for iteration $i = 1, 2 \ldots, I$ do
        Compute stochastic gradient $g_p^i$ on batch $\mathcal{B}_i$ of size $B_p$.
        Use (12) with the learning rate $\eta_p$ to project gradient onto
            coordinate wise constraint and update local model.
    end for
end for
Output: The update $g_p = w_p - w_s$.
```

In algorithm 1, I is the number of iterations per epoch, i.e. $I = \lfloor |\hat{D}|_p / B_p \rfloor$, and the update vector g_p will be sent to the server to poison the global model.

The convergence of PGD training can be obtained from the main results of [33]. This only needs to modify the gradient ascent of [33] to gradient descent, and then one can see that algorithm 1 can converge to the local optimal value of the problem (11). As for the convergence analysis of the federated learning with our backdoor attack, we show it in theorem 2, which relies on Assumptions 1 and 2 used in the standard analysis of SGD [34] and Assumption 3 commonly used in the federated optimization literature [35], [36], [37].

Assumption 1 (Smoothness): Local loss function of each user k is Lipschitz smooth, i.e.: $|\nabla l_k(w) - \nabla l(w')||_2 \leq L||w-w'||$.

Assumption 2 (Unbiased Gradient and Bounded Variance): The stochastic gradient at each user side is an unbiased estimator of the local gradient $E[g_k(w)] = \nabla l_k(w)$, and has bounded variance $E[||g_k(w) - \nabla l_k(w)||_2^2] \leq \sigma^2$.

Assumption 3 (Bounded Dissimilarity): For any sets of weights $\{\mu_k \geq 0\}_{k=1}^K$, where $\sum_k \mu_k = 1$, there exist constants γ^2 and κ^2 such that $\sum_k \mu_k ||\nabla l_k(w)||_2^2 \leq \gamma^2 || \sum_k \mu_k \nabla l_k(w)||_2^2 + \kappa^2$.

Our main theorem is stated as follow:

Theorem 2. *In the considered FL setting, assume the iterations of each user local training is* τ_k, $k \in \{1, 2, \cdots, K\}$. *Assume that the attacker participates in federated learning from the* t_0*-th round, and continues to participate in* T_p *rounds, and the loss of the global model* $f_{w^{(t_0,0)}}$ *on the dataset* $D_p \cup D_c$ *is* $\mathcal{L}(w^{(t_0,0)})$*, and the loss of the optimal backdoor is* $\mathcal{L}(w^{(*,0)}) \geq l_{min}$*. Then, we have:*

$$\frac{1}{T_p}\left(\mathcal{L}(w^{(t_{T_p},0)}) - \mathcal{L}(w^{(*,0)})\right) \leq \\ \frac{1}{T_p}\left(\mathcal{L}(w^{(t_0,0)}) - l_{\min} \\ -\eta_0 \lambda^{t_{T_p}} \left(\sum_k p_k \tau_k\right) \sum_{i=t_0}^{t_{T_p}} ||\nabla \mathcal{L}(w^i)||_2^2/8\right) + V,$$

where

$$V = \left(\sum_k p_k \tau_k\right) {\eta_0}^2 \lambda^{2t_0} \left[(L\sigma^2 + 3L\kappa^2)/|C| + 6\eta_0 \lambda^{t_0} \sigma^2 \\ +12\eta_0 \lambda^{t_0} \kappa^2 \tau_{\max}(\tau_{\max} - 1)\right].$$

and $\tau_{\max} = \max\{\tau_k\}$ *is the max value of the user local training iteration.*

Proof : Here, we briefly introduce the sketch of the proof, please refer to the detailed proof in the appendix C. In our FL setting, the datasets are unbalanced and non-iid across users, thus, the iterations of local-update stochastic gradient descent with a mini-batch size B of each user is different. The conclusions of the literature [38], [39], [40], [41], [42] on the convergence of FL are not applicable to the scenario we are discussing, because these literatures assume that the number of iterations updated locally by each user is the same. We adopt the convergence analysis framework proposed in [37] and follow the same assumptions as it to get the proof of the above theorem. The main idea is to first obtain the upper bound of the change in the loss function $\mathcal{L}$ under the strategy of randomly sampling $|C|$ users participating in communication , i.e. $\mathcal{L}(w^{(t+1,0)}) - l(w^{(t,0)}) \leq \mathcal{V}$ and then use the recursive method to obtain the result of theorem 2.

From theorem 2, we see that consistent with intuition, increasing the attack starting point t_0 will reduce the distance between the global model and the optimal backdoor model, i.e., attacking the model in the later stage of the FL training (at this time, the global model tends to converge) can make the backdoor attack more likely to succeed. This can be confirmed from the experimental results in Section IV.

It is worth noting that, even if the attack is successful, the global model has a backdoor, if the attacker stops attacking, will the backdoor have the same maintenance status for different target missions? In other words, if federated learning is still going on after the attacker leaves, will the global model have the same backdoor forgetting effect for different target subtasks? Here, we give the theoretical analysis, i.e., theorem 3. And the corresponding experimental results to verify the theory in Section IV shows that the answer is negative.

Theorem 3. *For the poison data* $x_p, y_p \in D_p$*, and the difference of the loss between a backdoor model* $w^{(t^*,0)}$ *and the model trained for* n *rounds without the attacker's participation starting from* t^* *is bounded by:*

$$\|\mathcal{L}(f_{w^{(t^*,0)}}(x_p), y_p; D_p \cup D_c) - \mathcal{L}(f_{w^{(t^*+n,0)}}(x_p), y_p; D_c)\|_2 \\ \leq \sum_{t=t^*}^{t^*+n} \|a_t g_t(x_p, y_p)\|_2, \tag{13}$$

where $w^{(t^*,0)}$ *is the backdoor model trained with* $D_p \cup D_c$*,* $w^{(t^*+n,0)}$ *is the model trained with* D_c *but start from* $w^{(t^*,0)}$*,* a_t *is a constant related to the current round of learning rate, and* g_t *is the gradient of the global model* $w^{(t,0)}$ *on data sample* (x_p, y_p)*.*

Proof : The proof of the above concise theorem can be found in the appendix D. The main idea is to first analyze the time derivative of the loss of the common example in the continuous domain, and then use the integral to obtain the change in the loss. The above theorem provides a way to measure the forgetting characteristics of the global model on the backdoor dataset.

Although we only introduce the backdoor solution of the system using FedAvg solution here, our work can also be applied to the other FL solution-based system, e.g. FedProx [43] solution. However, a detailed exploration of this is outside the scope of this paper, and we think we provide generalized ideas for backdoor attacking FL that are worthy of further exploration.

B. Our defense method

Defending backdoor attacks in the FL scenario refers to the neutralization of backdoor attacks while ensuring that the benign accuracy is not significantly reduced. Based on the intuition that the attacker using the global model to perform backdoor training will cause significant changes to the global model, norm-clipping defense (NCD) is proposed in [44]. According to this defense method, the server checks the norm difference between the global model shared to the users and the model updates sent back from the selected users and uses a pre-specified norm difference threshold to clip model updates that exceed the norm threshold. Specifically, NCD uses a pre-specified threshold ξ and the following formula to aggregate user-uploaded updates and get a new global model:

$$\begin{cases} \Delta^t = \sum_{i=1}^C p_i \frac{\Delta_i^t}{\max(1, ||\Delta_i^t||_2/\xi)}, \\ w^{(t+1,0)} = w^{(t,0)} + \Delta^t. \end{cases} \tag{14}$$

The above aggregate method (14) ensures that the norm of each model update is small, hence less susceptible to the server and limits the success of backdoor attacks.

Obviously, the above threshold ξ cannot be set too small. A small threshold will cause neither the benign models nor an attacker's model to pass the defense, i.e., all the updates will be bounded by ξ, which will result in a decrease in the convergence efficiency of the global model and a performance loss. This threshold also can't be set too large, a large threshold will cause the attacker to escape the defense. In addition, since the global model gradually converges during the FL training process, the update amount of the users' model will gradually become smaller. Therefore, fixing the above threshold ξ is not always the best choice. We propose a new defense method that dynamically adjusts this threshold this threshold. Specifically, the defense method we propose uses the following formula to calculate the amount of model update required by the server aggregation:

$$\Delta_d^t = \sum\nolimits_{i=1}^{C} p_i \frac{\Delta_i^t}{\max(1, ||\Delta_i^t||_2/\psi(\xi^0, t))}, \tag{15}$$

where $\psi(\cdot, t)$ is the update formula of the threshold ξ, which gradually decreases with the increase of t, and ξ^0 is the initial threshold. In this work, we set $\psi(\xi^0, t) = \xi^0 \mu^t$ and μ is a hyperparameter that controls the threshold decay rate.

C. *Our detection method*

The detection of a backdoor attack refers to distinguishing the benign models and the models with a backdoor in a given set of models (the detector does not know the attacker's attack method and attack target). Since a backdoored model behaves perfectly innocently during inference with benign inputs, the backdoor is difficult to be detected compared with classical byzantine attacks whose aim is to reduce a classifier's accuracy, and the poor performance is more easily detectable.

Inspired by the research on noise response analysis of nonlinear dynamic systems [45], [46], we propose a federated noise titration scheme to detect backdoors. In quantitative chemical analysis [47], the main insight of titration is to drop a reagent solution with a known accurate concentration into the solution of the test substance until the response is complete, and then determine the characteristics of the test substance according to the concentration of the reagent solution used. We add noise with a specific intensity (concentration) to the mean data of each user. This noisy input is regarded as a reagent solution, the concentration is the variance of the noise, and the test substance is the measured model. We measure the response of the model under different concentrations of noise to determine whether the model is infected by a backdoor attack. Specifically, for k-th user, we add i.i.d. normal-distributed noise $\delta \sim N(0,1)$ scaled by r (i.e., $r\delta \sim \mathcal{N}(0, r^2)$) to its input x_i. We measure the responses of the model $f_{w_k^{(t,0)}}$ with the noisy input $x_i + r\delta$, and then record the confidence of predictions π_r^χ which is defined as:

$$\pi_r^\chi = \frac{|\{\overline{y}_i^r > \chi\}_{i=1}^{N_k^r}|}{N_k^r}, \tag{16}$$

where $\overline{y}_i^r = \max \sigma(f_{w_k^{(t,0)}}(x_i + r\delta_i)), \delta_i \sim N(0,1)$ is the maximum output softmax activation σ, N_k^r is the number of sample noise, and χ is a tunable threshold used to distinguish high- and low-confidence predictions. The intuition to use the confidence (16) introduced above to distinguish whether the model is infected by a backdoor is that noise leads to logits of backdoored models to be very large, which yields high-confidence predictions. According to this intuition, we propose a federated noise titration method to efficiently detect the backdoor. Using algorithm 2, the server can record a list of confidences with different r. If the confidence π_r^χ rises rapidly as r increases and reaches 1, the global model is judged to be infected by a backdoor attack. Since algorithm 2 only requires users to perform forward propagation without updating the neural network, and the server only needs to count the results uploaded by users, the computational complexity is low. Moreover, the data uploaded by the user is only the logical outputs of the neural network, and the dimension is equal to $N_b = 32$, so the communication transmission cost is also low. Therefore, the above detection algorithm is efficient and has a low computational cost.

Algorithm 2 Federated noise titration

```
Require: Local dataset $D_k$, number of sampled noise $N_k^r$ and
    downloaded shared global model $w_s$.
for $r = r_1, r_2 \dots, r_{max}$ do
    for user $k = 1, 2 \dots, K$ do
        Sampling some inputs $\{x_i\}_{i=1}^{N_k^r}$.
        Calculates the softmax output:
        $\overline{y}_i^r = \max \sigma(f_{w_s}(x_i + r\delta_i)), \delta_i \sim N(0,1)$.
        Send $\{\overline{y}_i^r\}_{i=1}^{N_k^r}$ to the server.
    end for
    Server merges the received outputs and calculates the overall
    confidence:
        $\begin{cases} \pi_r^\chi = \frac{|D_r^\chi|}{\sum_{k=1}^{K} N_k^r}, \\ D_r^\chi = \underset{k=1,\dots,K}{\cup} \{\overline{y}_i^r > \chi\}_{i=1}^{N_k^r} \end{cases}, \quad (17)$
end for
```

IV. Empirical Evaluation

In this section, we show the performance of our proposed backdoor attack method through experiments. First, we show the effectiveness of our backdoor method (algorithm 1). Then, we verify the superiority of the proposed defense method. Besides, we compare the proposed backdoor implantation method with the classic model poisoning-based backdoor method (baseline) [16], and we show that our defense method can defend against both of these attacks. In addition, we verify the effectiveness of our backdoor detection method (algorithm 2).

The experiment setup is as follows: the mmWave channel is centered at 28GHz with 200MHz bandwidth and the noise power is set -12.6dB. The wireless environment simulation method is the same as [48]. We deploy base stations at different locations on 10 different streets. Assume that there are 10 users participating in federated learning on each street, each user has its own driving trajectory, and their local dataset is non-iid. The

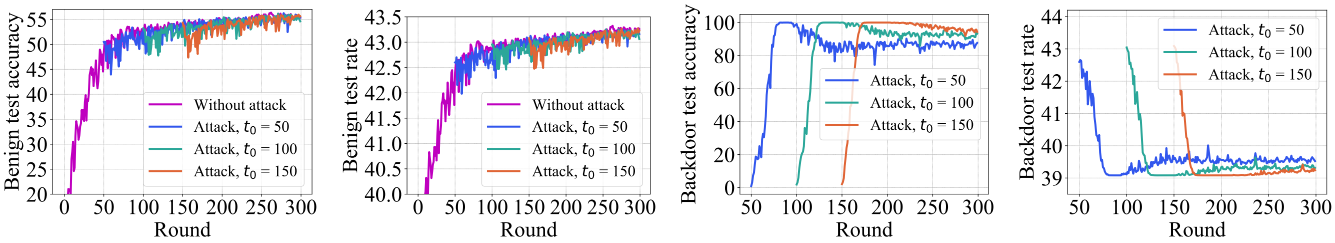


Fig. 2: *Performance of the first backdoor target sub-task.*

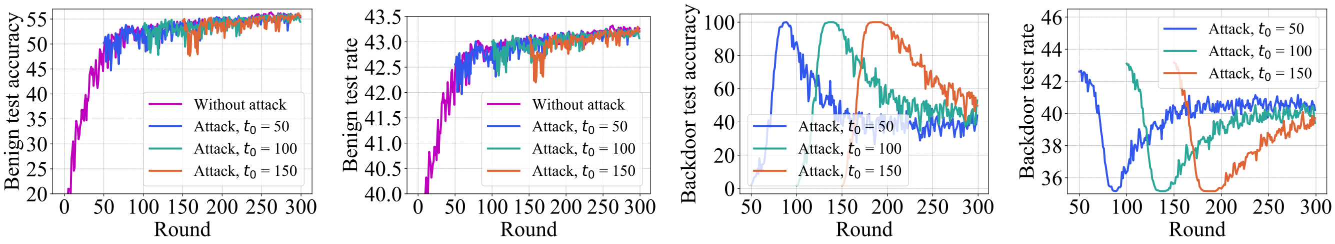


Fig. 3: *Performance of the second backdoor target sub-task.*

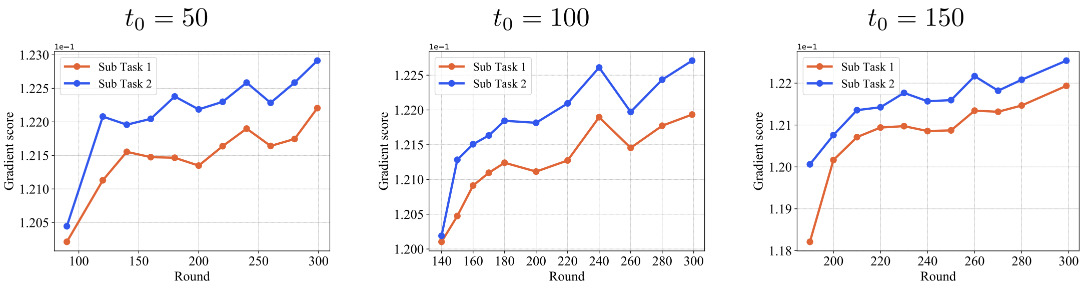


Fig. 4: *Gradient score of two target sub-tasks at different* t_0*.*

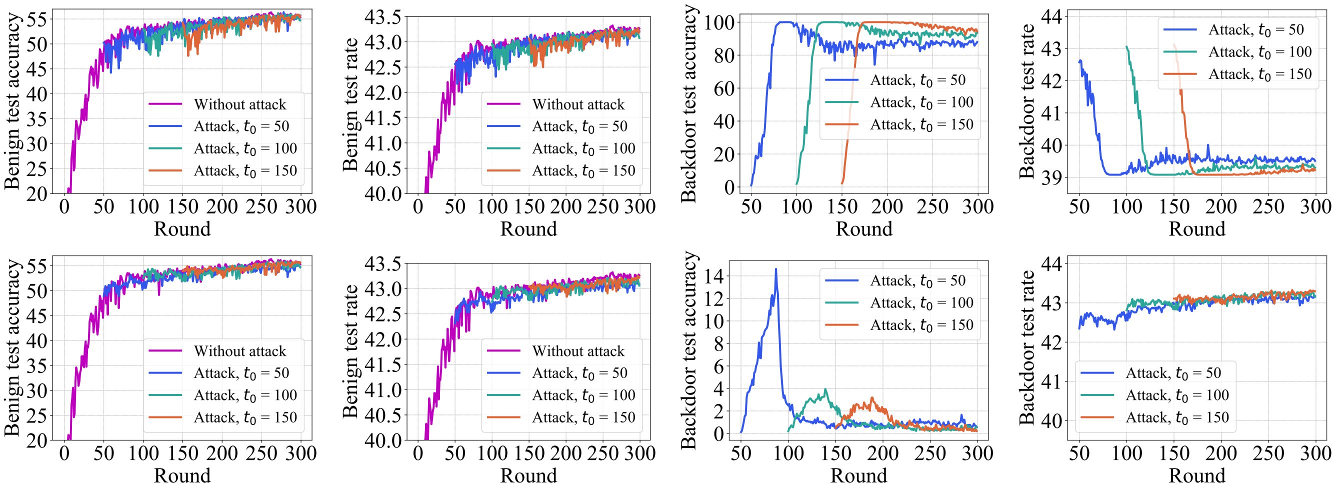


Fig. 5: *Performance of the first backdoor target sub-task with different defense method. (Top) NCD defense method. (Bottom) Our defense method.*

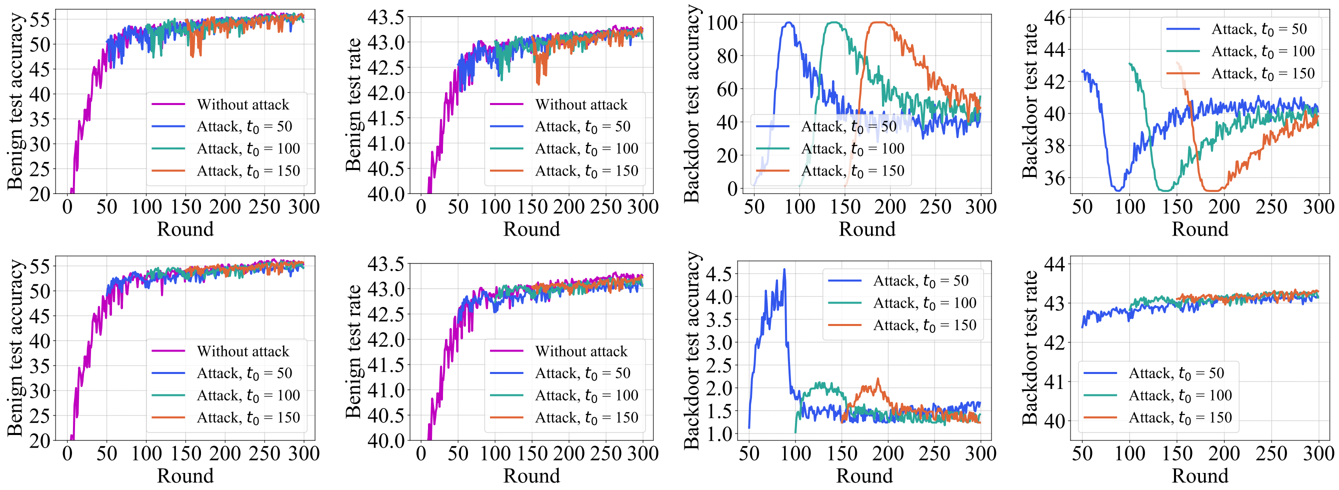


Fig. 6: *Performance of the second backdoor target sub-task with different defense method. (Top) NCD defense method. (Bottom) Our defense method.*

distribution of the label of each user's dataset can be seen in appendix. We use VGG16 [49] as the neural network structure, and modify the size of the fully connected layer and the output layer of the classifier to make the network adapt to our input and labels of the considered beam selection problem. Please see the final neural network structure is in the appendix A. The mini-batch size B of each user is 20. We use the SGD optimizer with 0.001 learning rate to perform local training, the initial learning of the server is $\eta_0 = 1.0$ and the learning decay parameter is $\lambda = 0.99$. In each round, the server randomly selects 10 users, i.e. $|C_t| = 10$ to participate in federated learning, and the entire training lasts for 300 rounds.

A. Backdoor beam selection system

In the experiment in this subsection, we set the initial rounds of the backdoor attack to 50, 100, and 150, and the attacker participates in 40 rounds (the attacker has a large number of local models, and each round of communication has a poisoned model selected). Fig. 2 and 3 show the performances of the backdoor attack on the two target sub-tasks. For each figure, from left to right we show the accuracy of the global model on the benign test data, the rate that the users can obtain when the backdoor is not triggered, the accuracy of the global model on the poisoned test data, and the rate of users obtained after the backdoor is triggered.

It can be seen that our attack method can successfully implant the backdoor into the global model during FL training. After the attacker participates in FL for a period of rounds, the global model can obtain 100% accuracy on the test poisoned dataset. In addition, because our attack did not cause a big shake to the test accuracy of benign data, our attack is concealed and difficult to be discovered. It is worth noting that after the attacker leaves, the global model still maintains a high accuracy on the poisoned data. For example, for target sub-task 1, the test accuracy of the global model on the poisoned dataset at the end of the FL training is than 80%. Besides, it can be seen that as t_0 increases, the effect of the backdoor becomes better.

However, it also can be found that for different target subtasks, the maintenance performance of the backdoor attack is different. Comparing Fig. 2 and Fig. 3, it can be found that the backdoor model seems to be easier to forget Task 2. We use theorem 3 to study the causes of the above phenomenon. Since we use the same learning rate for the two tasks, we can only show the gradient score $g_s = \sum_{t=t^*}^{t^*+n} \|g_t(x_p, y_p)\|_2$ to describe the forgetting curve of the model, i.e, Fig. 4. It can be seen that the gradient score of task 2 is always higher than that of task 1, so task 2 is more likely to be forgotten. In addition, it can be found that as the starting point moves backward, the gradient scores of the two tasks are reduced, which further proves that the attack in the later stage of the learning phase is more conducive to the survival of the backdoor.

B. Defense backdoor beam selection

Here, we show the superiority of our defense method through experiments. In the experiment, we still set the initial rounds of the backdoor attack to 50, 100, and 150, and the attacker participate in 40 rounds. Fig. 5 and 6 show the backdoor performance under NCD and our defense method.

It can be found that using the NCD method cannot effectively defend our backdoor attacks, which means that the global model still shows high accuracy on the poisoned test dataset. This is mainly because our attack method uses PGD to limit the distance between the poisoned model and the benign model. Therefore, the defense based on norm clipping fails. However, our proposed dynamic norm clipping algorithm dynamically reduces the threshold of norm clipping during the training process, making it difficult for the poisoning model to contribute to the global model, so it shows a good defense effect, i.e., the test accuracy of the global model on the poisoning dataset is very low, and the performance on the benign dataset is almost the same as the benign model.

C. Comparison with baseline backdoor implant method

Here, we compare our backdoor implant method with the baseline which just use model poisoning without PGD updating.

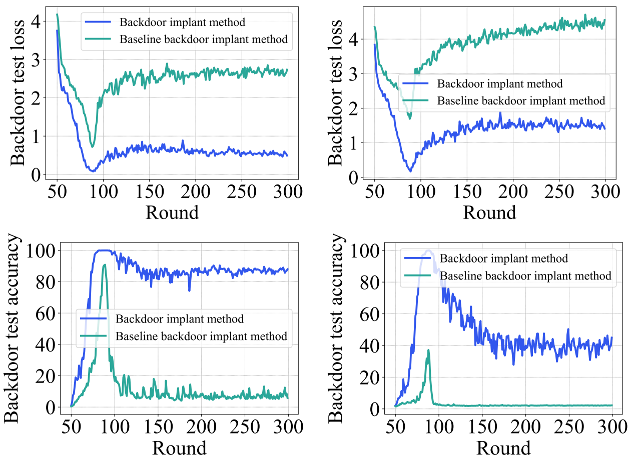


Fig. 7: *Comparison with baseline backdoor implant method on the first (left) and the second (right) target sub-tasks with NCD defense method. (Top) Backdoor test loss. (Bottom) Backdoor test accuracy.*

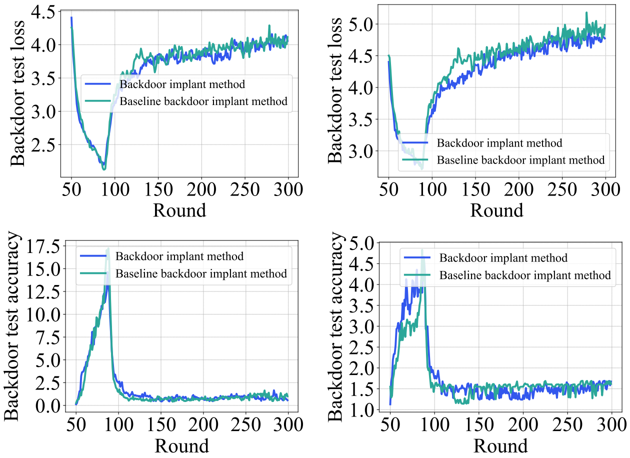


Fig. 8: *Comparison with baseline backdoor implant method on the first (left) and the second (right) target sub-tasks with our proposed defense method. (Top) Backdoor test loss. (Bottom) Backdoor test accuracy.*

We conduct the experiment on the setting of the initial round of the backdoor to 50. First, we show the backdoor test loss and accuracy on the setting of sever uses the NCD defense method. The results are shown in Fig. (7). One can see that, for both of these two tasks the baseline backdoor implant method cannot effectively implant the backdoor. Besides, after the attacker stopped the attack, our backdoor accuracy is still higher than the baseline. Second, we show the test performance of these two backdoor implant methods on the setting of sever uses our proposed defense method. The results are shown in Fig. (8). We can see that compared to using the NCD defense method, using our defense method can make the test losses of these two attacks are high, and the test accuracies are low, which means that these attacks can not successfully implant backdoors into the global model. Therefore, our defense method can effectively defend against various backdoor attacks.

D. Backdoor beam selection with only one attack device

In this section, we discuss the performances of backdoor attacks and defenses when the attacker can only control one device. This means that only if this only one attacker is selected by the server which uses a randomly sampling policy for selecting users, the attacker can send the poisoned model to the server.

The results of the backdoor attack and defense of the two considered tasks under the above setting are shown in Fig. 9 and 10. It can be found that when there is no defense or the NCD method is used, our backdoor attack is still successful in implanting the backdoor into the global model at the end of the training. Fortunately, the defense method we proposed effectively blocked the backdoor attack, but judging from the test accuracy curve of the benign data of task 1 in Fig. 9, our defense algorithm slightly sacrificed the convergence rate.

E. Detection of backdoor model

Although we have demonstrated the effectiveness of the proposed defense method above, detection is still valuable because we believe that there may still be more powerful backdoor attacks that can bypass our defense method. In this subsection, we show the performance of our detection method in identifying the benign model and backdoor model.

We use circle plots, i.e., Fig. 11 in which each class is represented by a colored section of a ring to visualize the effectiveness of the proposed federated noise titration method. Each point within the circle is an output $z(x,r) = f_{w_s}(x_i+r\delta_i)$ represented by a polar coordinate. The logit $z(x,r)$ is a vector of dimension N_b, i.e., $z(x,r) = [z_1, z_2, \cdots, z_{N_b}]$. The radius of each point is $max\{z_1, z_2, \cdots, z_{N_b}\}$, i.e., the maximum logit. The angle is given by $\frac{\arg\max\{z_1,z_2,\cdots,z_{N_b}\}}{2\pi N_b}$. The radius indicates the level of confidence. From Fig. 11, we see that if we increase the input noise intensity (left to right), one observes high-confidence predictions for all noisy data of the backdoor model. It can be found that as r increases, compared with the benign model, the backdoor model quickly predicts all noise samples into one class. Thus, we can use this phenomenon to screen whether the global model is a backdoor model.

In particular, we show the changing trend of the confidence of the two kinds of the model as the threshold χ changes when r takes different values. From Fig. 12 we see that as r increases, the threshold required for the confidence to decay to 0 becomes larger. In other words, by choosing an appropriate threshold χ, the confidence of the backdoor model will reach 1 faster than the benign model. We choose $\chi = 0.7$ as the final threshold. As can be seen from the results in Fig. 13, the confidence of the backdoor model will increase rapidly, while the confidence of the benign model will increase slowly. Thus, the proposed federated noise titration method can be used for efficient backdoor detection.

V. Conclusion

We have proposed a backdoor attack method for federated learning-based mmWave beam selection, where the attacker uses obstacles at specific locations on the road as a trigger set. We defined two types of tasks for attackers with specific

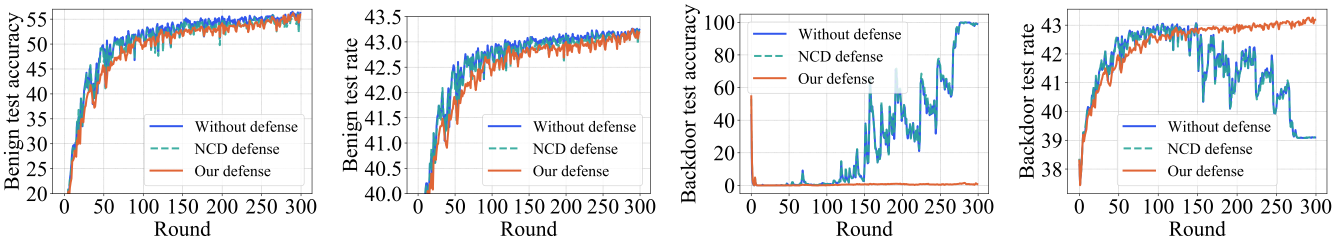


Fig. 9: *Performance of the first backdoor target sub-task with only one attacker.*

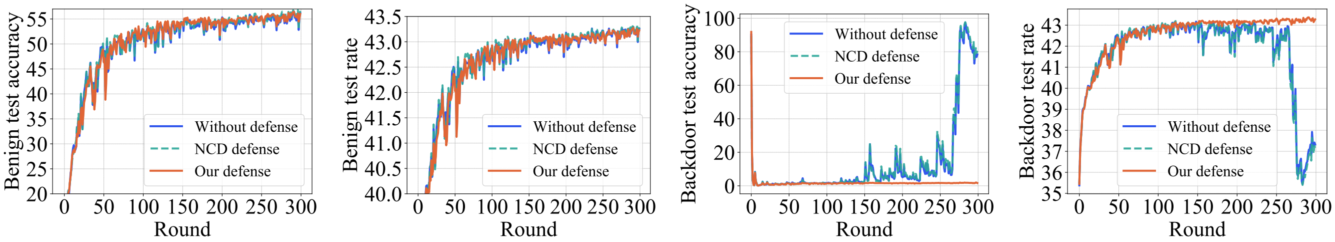


Fig. 10: *Performance of the second backdoor target sub-task with only one attacker.*

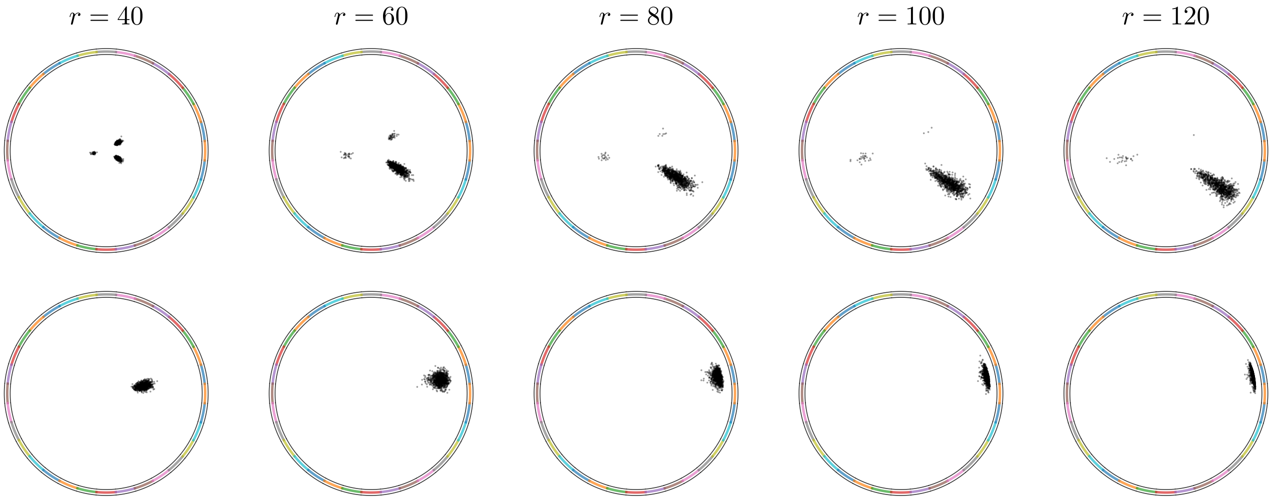


Fig. 11: *Circle plots of our federated noise titration solotion for benign model (Top) and backdoor model (Bottom).*

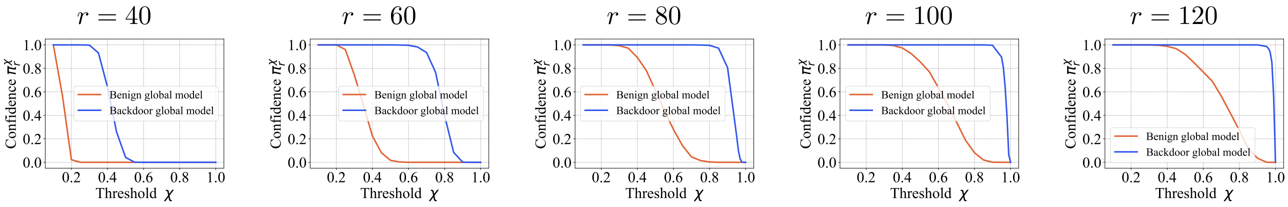


Fig. 12: *Titration curves vs. threshold χ for benign model and backdoor model with different noise intensity r.*

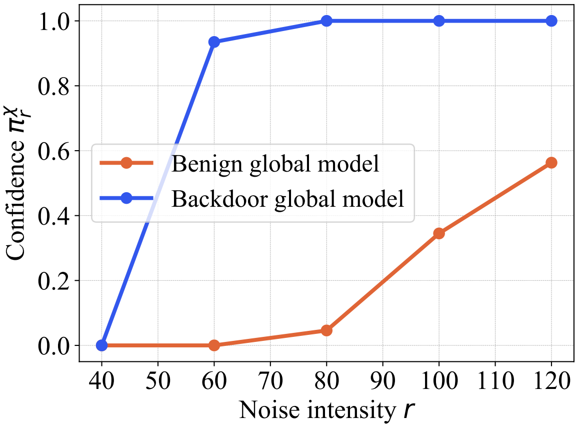


Fig. 13: *Confidence π_r^χ vs. noise intensity r of benign model and backdoor model with threshold $\chi = 0.7$.*

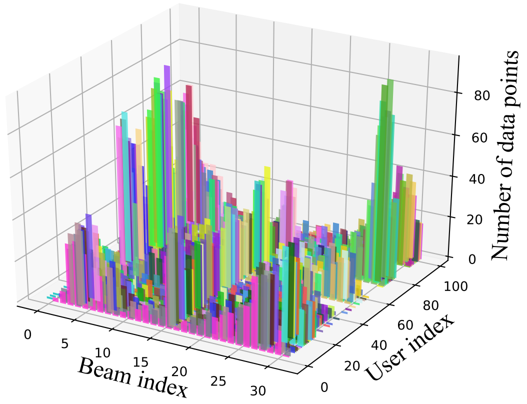


Fig. 14: *Vis Non-iid distribution*

purposes, the first one is to let the model that triggered the backdoor output a specific beam, and the second is to let the backdoor model output the beam that minimizes the average user rate. We gave the theory of the existence of the backdoor model and proposed a simple and effective backdoor implantation algorithm. We studied our backdoor attack method under the system without defense and with norm clipping defense, and the results showed that our backdoor attack can be successful in both of the above cases. Then, we proposed an effective defense method, i.e., dynamic norm clipping defense, to make this attack fail and ensure the normal operation of the beam selection system. We also proposed a backdoor detection method that used noise response analysis to efficiently distinguish the backdoor model and the benign model. Our extensive empirical evaluation demonstrated there was a risk of simple backdoor attacks in the beam selection system based on federated learning. Defense and detection methods are all required to ensure the high reliability of the system. We believe that our work can inspire more research on robust machine learning-based wireless communication.

Appendix A
Deep Neural Network, Computing infrastructure and Non-iid Dataset Distribution Visualization

We use the neural network architecture shown in Tab. I to be the local model of each user. In this table $Channel$ is the channel number, ks is the kernel size, s is stride and pad is the padding of the Conv2d layer. eps and momentum are the hyperparameters of the Batch Normalization (BN) layer.

All experiments use NVIDIA GPU servers as computing nodes. Each server contains 2 Tesla P100 GPUs and the servers are internet worked via commodity 1 Gbps Ethernet. The experiments are implemented in PyTorch, and we generate the weights of the neural networks of each user with random seed 1. Thus, we can simulate broadcasting the same initialized parameters to all users, and the results are reproducible.

We count the distribution histograms of data labels on different user devices to visualize the distribution of users' datasets. From Fig. 14, one can clearly see three characteristics. First, for each user, the dataset is unbalanced i.e., the number of samples in each class is not the same. Second, the amount of the data owned by different users may be different, which implies that the number of iterations for the users training their local model per epoch may be different. Third, there is a label skew between users, i.e., the label distribution of different users may be different. The above three characteristics indicate the datasets across users are non-iid.

Appendix B
Proof of the existence of the backdoor, Theorem 1

Here we give a proof of Theorem 1. First, the following lemma show the existence of adversarial examples on cube:

Lemma 1 *Assume for the classification problem, each class c distributed over the unit hypercube $[0,1]_n$ with density functions $\{P_c\}_{c=1}^{C}$. For a classifier that partitions the hypercube into disjoint measurable subsets. Let U_c denote the supremum of P_c. Let $Ratio_c$ be the fraction of hypercube partitioned into class c by C. Choose some class c with $Ratio_c \leq 0.5$, and select an ‘p-norm with $p > 0$. Define $p^* = min(p, 2)$. Sample a random data point x from the class distribution P_c, Assuming it can be correctly classified by the classifier. Then with probability at least $1 - U_c \exp(-\pi n^{1-2/p^*} \varepsilon^2)/(2\pi n^{1/2-1/p^*})$, x has an adversarial example x' with $||x - x'||_p \leq \varepsilon$.*

The proof of Lemma 1 can be seen from [50]. Recall that in the data preprocessing, the data will be processed into a 2-dimensional matrix, and the value is limited to the range of 0-1, so we can get the probability that the existence of adversarial examples from the above lemma.

Since our trigger set is obstacles at certain locations on the road, obviously after data preprocessing, these obstacles will appear in the poisoned data in the form of patterns. At the same time, since the appearance of these obstacles is rare in benign samples, poisoned samples are edge cases. That is, our trigger set has the duality of pattern and edge case. Thus, combining the introduction of Lemma 1 about the existence of adversarial samples and the conclusion of [51] about the connection between adversarial examples and backdoor, we can

Table I: *Deep neural network model used for FL-based beam selection.*

Layer	Activation	Hyperparameters
Conv2d		$Chaneel = 64$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 64$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 64$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 128$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 128$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 256$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 256$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 256$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 512$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 512$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 512$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 512$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 512$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
Conv2d		$Chaneel = 512$, $ks = (3,3)$, $s = (1,1)$, $pad = (1,1)$
BN	ReLU	$eps = 1e-5$, momentum=0.1
MaxPool2d		$ks = (2,2)$, $s = 2$, $pad = 0$
AvgPool2d		$ks = (1,1)$, $s = 1$, $pad = 0$
Dense		Number of neure 4608, output size 32

get the conclusion about the existence probability of backdoor in Theorem 1.

For a benign baseline model f_w and a perturbed model $f_{w'}$, assume the weights of their l-th layer are w_l and w_l'. Considering the following equations:

$$\begin{array}{l} w_l x_j^l = w_l^{'} x_j^l, x_j \in D_c \\ w_l {x'}_j^l = w_l^{'} x_j^l, x_j \in D_p, \end{array} \tag{18}$$

where x' is adversarial example of x with $||x - x'||_2 \leq \varepsilon$. One can see that:

$$\begin{array}{l} (w_l - w_l')x_j^l = 0, x_j \in D_c \\ (w_l - w_l')x_j^l = w_l \varepsilon_j^l, x_j \in D_p \end{array}, \tag{19}$$

where ε_j^l is the l-th layer feature of the adversarial perturbation of x_j. Let X_l be the activation matrix of l-th layer, we can rewrite (19) as:

$$(w_l - w_l')X_l^T = w_l E_l, \tag{20}$$

where E_l is the matrix of adversarial perturbations. Recall we assume invertibility of $X_l X_l^T$, thus, we get:

$$w_l - w_l' = w_l E_l (X_l X_l^T)^{-1} X_l. \tag{21}$$

Following Lemma 1, and using the same way of [51], one can recover inequality in Theorem 1.

Appendix C
Proof of the convergence analysis of backdoor Federated learning, theorem 2

First, according to [37], if the server random select $|C|$ user model and uses η_t as the learning rate in round t, we get:

$$\begin{array}{l} L(w^{(t+1,0)}) - L(w^{(t,0)}) \leq \\ \eta_t \left(\sum_k p_k \tau_k\right) \Big[- \left\|\nabla L(w^{(t,0)})\right\|_2^2 /8 + \frac{\eta_t \left(\sum_k p_k \tau_k\right)\left(L\sigma^2 + 3L\kappa^2\right)}{|C|} \\ +6\eta_t^2 L^2 \sigma^2 \frac{\sum_k p_k \tau_k (\tau_k - 1)}{\sum_k p_k \tau_k} + 12\eta_t^2 L^2 \kappa^2 \tau_{\max}(\tau_{\max} - 1) \Big] , \end{array} \tag{22}$$

let $L(w^{(t+1,0)}) - L(w^{(t,0)}) \leq \psi(\eta_t, \left\|\nabla L(w^{(t,0)})\right\|_2^2)$, we get:

$$\begin{array}{l} L(w^{(t_{T_p},0)}) - L(w^{(t_0,0)}) = L(w^{(t_{T_p},0)}) - L(w^{(t_{T_p}-1,0)}) \\ +L(w^{(t_{T_p}-1,0)}) - L(w^{(t_{T_p}-2,0)}) \\ + \cdots + L(w^{(t_0+1,0)}) - L(w^{(t_0,0)}) \\ \leq \sum_{t=t_0}^{t_{T_p}} \psi(\eta_t, \left\|\nabla L(w^{(t,0)})\right\|_2^2). \end{array} \tag{23}$$

Recall we assume the optimal backdoor is $\mathcal{L}(w^{(*,0)}) \geq l_{min}$, and $\eta_t = \eta_0 \lambda^t$, one can quickly get the bound shown in Theorem 2.

Appendix D
Proof of the upper bound of the loss change, theorem 3

Here, first, we give the time derivative of the loss for a generic labeled example (x_p, y_p):

$$\frac{dL(f_w(x_p), y_p)}{dt} = g_t(x_p, y_p)\frac{dw_t}{dt}, \tag{24}$$

where g_t is the gradient of the global model w_t on data sample (x_p, y_p). To deal with the above differential, we use the following discrete-time dynamics:

$$\frac{dw_t}{dt} \approx w_{t+1} - w_t = -\frac{\eta_t}{|D_t|} \sum\nolimits_{(x,y) \in D_t} g_t(x,y). \tag{25}$$

According to the convergence analysis of the global model, the norm of the gradient of the global model is bounded, assume $\eta_t \left\| \frac{d\mathcal{L}(f_{w_t}(x), y)}{dw_t} \right\|_2 \leq c_t$, then we get:

$$\begin{array}{l} \|\mathcal{L}(f_{w^{(t^*,0)}}(x_p), y_p; D_p \cup D_c) - \mathcal{L}(f_{w^{(t^*+n,0)}}(x_p), y_p; D_c)\|_2 \\ \leq \|\mathcal{L}(f_{w^{(t^*+1,0)}}(x_p), y_p; D_c) - \mathcal{L}(f_{w^{(t^*,0)}}(x_p), y_p; D_p \cup D_c)\|_2 \\ + \cdots + \\ \|\mathcal{L}(f_{w^{(t^*+n,0)}}(x_p), y_p; D_c) - \mathcal{L}(f_{w^{(t^*+n-1,0)}}(x_p), y_p; D_c)\|_2 \\ \leq \int_{t^*}^{t^*+1} \left\| \frac{d\mathcal{L}(f_w(x_p), y_p; D_c)}{dt} - \frac{d\mathcal{L}(f_w(x_p), y_p; D_p \cup D_c)}{dt} \right\|_2 dt \\ + \cdots + \\ \int_{t^*+n-1}^{t^*+n} \left\| \frac{d\mathcal{L}(f_w(x_p), y_p; D_c)}{dt} - \frac{d\mathcal{L}(f_w(x_p), y_p; D_p \cup D_c)}{dt} \right\|_2 dt \\ \leq \sum_{t=t^*}^{t^*+n} c_t \|g_t(x_p, y_p)\|_2 \end{array} \tag{26}$$

References

[1] M. Alrabeiah, A. Hredzak, and A. Alkhateeb, "Millimeter wave base stations with cameras: Vision-aided beam and blockage prediction," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 2020, pp. 1–5.

[2] G. Charan, M. Alrabeiah, and A. Alkhateeb, "Vision-aided 6g wireless communications: Blockage prediction and proactive handoff," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10 193–10 208, 2021.

[3] J. Zhang, Y. Huang, Y. Zhou, and X. You, "Beam alignment and tracking for millimeter wave communications via bandit learning," *IEEE Transactions on Communications*, vol. 68, no. 9, pp. 5519–5533, 2020.

[4] J. Zhang, Y. Huang, J. Wang, X. You, and C. Masouros, "Intelligent interactive beam training for millimeter wave communications," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 2034–2048, 2021.

[5] J. Wang, Z. Lan, C. woo Pyo, T. Baykas, C. sean Sum, M. Rahman, J. Gao, R. Funada, F. Kojima, H. Harada, and S. Kato, "Beam codebook based beamforming protocol for multi-gbps millimeter-wave wpan systems," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 8, pp. 1390–1399, 2009.

[6] J. Wang, Z. Lan, C.-S. Sum, C.-W. Pyo, J. Gao, T. Baykas, A. Rahman, R. Funada, F. Kojima, I. Lakkis, H. Harada, and S. Kato, "Beamforming codebook design and performance evaluation for 60ghz wideband wpans," in *2009 IEEE 70th Vehicular Technology Conference Fall*, 2009, pp. 1–6.

[7] C. Antón-Haro and X. Mestre, "Learning and data-driven beam selection for mmwave communications: An angle of arrival-based approach," *IEEE Access*, vol. 7, pp. 20 404–20 415, 2019.

[8] M. Alrabeiah and A. Alkhateeb, "Deep learning for mmwave beam and blockage prediction using sub-6 ghz channels," *IEEE Transactions on Communications*, vol. 68, no. 9, pp. 5504–5518, 2020.

[9] A. Klautau, P. Batista, N. González-Prelcic, Y. Wang, and R. W. Heath, "5g mimo data for machine learning: Application to beam-selection using deep learning," in *2018 Information Theory and Applications Workshop (ITA)*, 2018, pp. 1–9.

[10] S. Stalla-Bourdillon, H. Pearce, and N. Tsakalakis, "The gdpr: A game changer for electronic identification schemes? the case study of gov.uk verify," *Computer Law & Security Review*, vol. 34, no. 4, pp. 784–805, 2018.

[11] J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[12] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. A. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*, 2017, pp. 1273–1282.

[13] J. Konečnỳ, B. McMahan, and D. Ramage, "Federated optimization: Distributed optimization beyond the datacenter," *arXiv preprint arXiv:1511.03575*, 2015.

[14] A. M. Elbir and S. Coleri, "Federated learning for hybrid beamforming in mm-wave massive mimo," *IEEE Communications Letters*, vol. 24, no. 12, pp. 2795–2799, 2020.

[15] M. B. Mashhadi, M. Jankowski, T.-Y. Tung, S. Kobus, and D. Gündüz, "Federated mmwave beam selection utilizing lidar data," *IEEE Wireless Communications Letters*, vol. 10, no. 10, pp. 2269–2273, 2021.

[16] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in *AISTATS*, 2020, pp. 2938–2948.

[17] Y. E. Sagduyu, R. A. Berryt, and A. Ephremidesi, "Wireless jamming attacks under dynamic traffic uncertainty," in *8th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, 2010, pp. 303–312.

[18] Y. Lin, R. Zhang, L. Yang, and L. Hanzo, "Secure user-centric clustering for energy efficient ultra-dense networks: Design and optimization," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 7, pp. 1609–1621, 2018.

[19] Y.-S. Shiu, S. Y. Chang, H.-C. Wu, S. C.-H. Huang, and H.-H. Chen, "Physical layer security in wireless networks: a tutorial," *IEEE Wireless Communications*, vol. 18, no. 2, pp. 66–74, 2011.

[20] K. Pelechrinis, M. Iliofotou, and S. V. Krishnamurthy, "Denial of service attacks in wireless networks: The case of jammers," *IEEE Communications Surveys Tutorials*, vol. 13, no. 2, pp. 245–257, 2011.

[21] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming sensor networks: attack and defense strategies," *IEEE Network*, vol. 20, no. 3, pp. 41–47, 2006.

[22] Y. E. Sagduyu, Y. Shi, and T. Erpek, "Adversarial deep learning for over-the-air spectrum poisoning attacks," *IEEE Transactions on Mobile Computing*, vol. 20, no. 2, pp. 306–319, 2021.

[23] Y. Shi, T. Erpek, Y. E. Sagduyu, and J. H. Li, "Spectrum data poisoning with adversarial deep learning," in *MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM)*, 2018, pp. 407–412.

[24] B. Kim, Y. Shi, Y. E. Sagduyu, T. Erpek, and S. Ulukus, "Adversarial attacks against deep learning based power control in wireless communications," *arXiv preprint arXiv:2109.08139*, 2021.

[25] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *CoRR*, vol. abs/1412.6572, 2015.

[26] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: A simple and accurate method to fool deep neural networks," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2574–2582, 2016.

[27] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *ArXiv*, vol. abs/1706.06083, 2018.

[28] J. Yang, B. Ai, K. Guan, D. He, X. Lin, B. Hui, J. Kim, and A. Hrovat, "A geometry-based stochastic channel model for the millimeter-wave band in a 3gpp high-speed train scenario," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 3853–3865, 2018.

[29] M. E. Morocho-Cayamcela, H. Lee, and W. Lim, "Machine learning for 5g/b5g mobile and wireless communications: Potential, limitations, and future directions," *IEEE Access*, vol. 7, pp. 137 184–137 206, 2019.

[30] A. Klautau, N. González-Prelcic, and R. W. Heath, "Lidar data for deep learning-based mmwave beam-selection," *IEEE Wireless Communications Letters*, vol. 8, no. 3, pp. 909–912, 2019.

[31] A. Shafahi, W. Ronny Huang, C. Studer, S. Feizi, and T. Goldstein, "Are adversarial examples inevitable?" *arXiv e-prints*, p. arXiv:1809.02104, Sep. 2018.

[32] H. Wang, K. K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J. yong Sohn, K. Lee, and D. Papailiopoulos, "Attack of the tails: Yes, you really can backdoor federated learning," *ArXiv*, vol. abs/2007.05084, 2020.

[33] Z. Deng, H. He, J. Huang, and W. J. Su, "Towards understanding the dynamics of the first-order adversaries," in *ICML*, 2020.

[34] L. Bottou, F. E. Curtis, and J. Nocedal, "Optimization methods for large-scale machine learning," *ArXiv*, vol. abs/1606.04838, 2018.

[35] J. Wang, A. K. Sahu, Z. Yang, G. Joshi, and S. Kar, "Matcha: Speeding up decentralized sgd via matching decomposition sampling," *2019 Sixth Indian Control Conference (ICC)*, pp. 299–300, 2019.

[36] S. J. Reddi, Z. B. Charles, M. Zaheer, Z. Garrett, K. Rush, J. Konecný, S. Kumar, and H. B. McMahan, "Adaptive federated optimization," *ArXiv*, vol. abs/2003.00295, 2021.

[37] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *ArXiv*, vol. abs/2007.07481, 2020.

[38] S. U. Stich, "Local sgd converges fast and communicates little," *arXiv preprint arXiv:1805.09767*, 2018.

[39] F. Zhou and G. Cong, "On the convergence properties of a k-step averaging stochastic gradient descent algorithm for nonconvex optimization," *arXiv preprint arXiv:1708.01012*, 2017.

[40] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of fedavg on non-iid data," *arXiv preprint arXiv:1907.02189*, 2019.

[41] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, 2019.

[42] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. Stich, "A unified theory of decentralized sgd with changing topology and local updates," in *International Conference on Machine Learning*. PMLR, 2020, pp. 5381–5393.

[43] A. K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. S. Talwalkar, and V. Smith, "On the convergence of federated optimization in heterogeneous networks," *ArXiv*, vol. abs/1812.06127, 2018.

[44] Z. Sun, P. Kairouz, A. T. Suresh, and H. B. McMahan, "Can you really backdoor federated learning?" *ArXiv*, vol. abs/1911.07963, 2019.

[45] M. T. Rosenstein, J. J. Collins, and C. J. De Luca, "A practical method for calculating largest lyapunov exponents from small data sets," *Physica D: Nonlinear Phenomena*, vol. 65, no. 1, pp. 117–134, 1993.

[46] S. Poon and M. Barahona, "Titration of chaos with added noise," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 98, pp. 7107–12, 07 2001.

[47] D. L. Reger, S. L. Goode, and D. W. Ball, "Chemistry: Principles and practice, 3rd ed." *yale journal of biology medicine*, 2009.

[48] A. Ali, N. González-Prelcic, and R. W. Heath, "Millimeter wave beam-selection using out-of-band spatial information," *IEEE Transactions on Wireless Communications*, vol. 17, no. 2, pp. 1038–1052, 2018.

[49] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv:1409.1556*, 2014.

[50] A. Shafahi, W. R. Huang, C. Studer, S. Feizi, and T. Goldstein, “Are adversarial examples inevitable?” *ArXiv*, vol. abs/1809.02104, 2019.

[51] H. Wang, K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J.-y. Sohn, K. Lee, and D. Papailiopoulos, “Attack of the Tails: Yes, You Really Can Backdoor Federated Learning,” *arXiv e-prints*, p. arXiv:2007.05084, Jul. 2020.