# Distributed Attacks over Federated Reinforcement Learning-enabled Cell Sleep Control

Han Zhang\*, Hao Zhou\*, Medhat Elsayed<sup>†</sup>, Majid Bavand<sup>†</sup>, Raimundas Gaigalas<sup>†</sup>,

Yigit Ozcan<sup>†</sup> and Melike Erol-Kantarci<sup>\*</sup>, Senior Member, IEEE

\* School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, Canada

<sup>†</sup> Ericsson Inc., Ottawa, Canada

{hzhan363, hzhou098, melike.erolkantarci}@uottawa.ca,

{medhat.elsayed, majid.bavand, raimundas.gaigalas, yigit.ozcan}@ericsson.com

Abstract-Federated learning (FL) is particularly useful in wireless networks due to its distributed implementation and privacy-preserving features. However, as a distributed learning system, FL can be vulnerable to malicious attacks from both internal and external sources. Our work aims to investigate the attack models in a FL-enabled wireless networks. Specifically, we consider a cell sleep control scenario, and apply federated reinforcement learning to improve energy-efficiency. We design three attacks, namely free rider attacks, Byzantine data poisoning attacks and backdoor attacks. The simulation results show that the designed attacks can degrade the network performance and lead to lower energy-efficiency. Moreover, we also explore possible ways to mitigate the above attacks. We design a defense model called refined-Krum to defend against attacks by enabling a secure aggregation on the global server. The proposed refined-Krum scheme outperforms the existing Krum scheme and can effectively prevent wireless networks from malicious attacks, improving the system energy-efficiency performance.

*Index Terms*—Federated learning, deep reinforcement learning, security, radio access networks, attacks, defense.

#### I. INTRODUCTION

With the deployment of the 5G and beyond 5G (B5G) networks, the increasing traffic demand for cellular communications has reached an unprecedented level [1]. To meet diverse service requirements and facilitate intelligent wireless communications, various machine learning (ML) techniques have been used to solve problems in wireless networks [2].

Reinforcement learning (RL) is a widely applied ML technique that provides automated solutions for high-complexity optimization problems [3]. Meanwhile, federated learning (FL) is another emerging ML technique that enables collaborative learning with local training in distributed systems, without sharing data. Federated reinforcement learning (FRL) is proposed as a combination of FL and RL and has proven effective in many wireless communication scenarios. For example, in [4], FRL is used to allocate power resources and radio resources in network slicing. However, these achievements of using FRL are mainly accomplished in fully secure environments without considering malicious attacks.

Due to the inherently distributed implementation, FL is more vulnerable to malicious attacks than other centralized ML techniques. Distributed participants in FL are easier to be attacked and manipulated, and the parameter sharing and updating between local and global servers may expose the FL to potential risks [5]. As a result, it is crucial to investigate security issues in FL.

There are some existing studies about attacks and defenses for FL algorithms [6]. However, most research focuses on supervised learning and cannot apply to FRL models. In this work, we study the security problem in an FRL-enabled cell sleep control scenario. As the traffic load grows, improving network energy-efficiency and reducing energy costs become critical goals for wireless networks [7]. Performing sleep control to base stations (BS) to reduce energy consumption is a feasible way to improve energy-efficiency and make networks sustainable [8]. However, attacks on cell sleep control may cause different levels of system performance degradation. For example, it may waste system energy by making BSs never sleep or produce low throughput by keeping BSs in sleep mode.

In this paper, we first design an FRL-based cell sleep control scenario and BSs will cooperatively learn sleep control strategies through FL. Then we assume some BSs are malicious participants. Specifically, we propose three attack models, namely free rider, Byzantine data poisoning, and backdoor attacks specifically for the given cell sleep control scenario. To the best of our knowledge, this is the first work that applies the backdoor attacks to a wireless network control application. The simulation results show that the designed attacks will lower system energy-efficiency. Meanwhile, we also propose a defense scheme called refined-Krum to defend against these attacks. Compared with the existing Krum defense scheme, it can achieve a better defense effect without knowing the number of attackers.

The rest of the paper is organized as follows. Section II introduces related works, and Section III shows our system model. Section IV introduces FRL-based sleep control scenario, and Section V presents the designed attacks and the proposed defense model. Finally, Section VI shows simulation results, and Section VII concludes this work.

#### **II. RELATED WORKS**

There have been many studies that design attacks and defenses towards breaches in FL algorithms. In [10], data poisoning attacks are performed on FL-based image classification problems. [11] performs backdoor attacks on the



Fig. 1. System Model.

FL system with single or multiple malicious participants. [9] proposes secure aggregation methods to defend Byzantine data poisoning attacks in the FL system. These works are only designed for supervised learning and do not apply to RL models. [12] and [13] proposes data poisoning attacks and defenses for FRL. However, these works are only tested with ready-to-use data sets and have some limitations if applied to complicated wireless network scenarios.

Meanwhile, other works study attacks on FL in wireless networks. [14] designs attacks specific to the wireless traffic prediction models in centralized and distributed scenarios. However, this work also uses a supervised learning model, and its attack method cannot be directly applied to other wireless network control applications that typically use RL. In [15], over-the-air jamming attacks on the uplink and downlink of FL in wireless networks are studied. But it only focuses on external attacks and fails to study the internal attacks in FL.

There are also some studies on cell sleep control for energy saving. [8] improves energy-efficiency of small cell networks by switching BSs to different modes. In [16], an RLbased traffic adaptive sleep mode control algorithm for BSs is proposed. However, these works are accomplished in fully secure environments and fail to consider attacks and defense. Different from existing studies, our work designs attacks to the specific FRL-based cell sleep control scenario. We focus more on vulnerabilities of FRL models related to realistic wireless environments and evaluate the effectiveness of attacks based on wireless network performance metrics.

#### **III. SYSTEM MODEL**

The system model is shown in Fig. 1. We consider a heterogeneous cellular network consisting of multiple BSs. There is one macro BS (MBS) cell and N small BS (SBS) cells cooperatively serving M distributed user equipment (UE) and handling traffic loads. The MBS is always active to ensure coverage and is responsible for controlling data services.

To effectively save energy costs of the system, we adopt three different sleeping modes for SBS cells, which are active, sleep, and deep sleep [8]. Active means SBSs are in full operation and consume the most energy. Sleep means SBSs temporarily stop transmitting data for the UEs but can be easily waken up and decide whether to continue sleeping in the next iteration. Deep sleep means more components are deactivated to save more energy, and SBSs take longer time to wake up. If a SBS turns to sleep, the arriving traffic will be offloaded to the MBS. SBSs sleeping at inconvenient intervals can cause low energy efficiency or data congestion in MBS traffic, thus degrading system performance.

This scenario considers a downlink orthogonal frequencydivision multiplexing cellular system. The link capacity between the  $m^{th}$  UE and the  $n^{th}$  SBS can be given as follows:

$$C_{n,m} = \delta_n \sum_{r \in R_n} B_r log_2(1 + SINR_{n,m,r}), \qquad (1)$$

where  $\delta_n$  is a binary indicator to denote whether the  $n^{th}$  SBS is active or sleeping.  $R_n$  denotes the set of available resource blocks of the  $n^{th}$  SBS and  $B_r$  denotes the bandwidth of the  $r^{th}$  resource block.  $SINR_{n,m,r}$  denotes the signal to interference noise ratio (SINR) between the  $m^{th}$  UE and the  $n^{th}$  SBS on the  $r^{th}$  resource block, which can be given as:

$$SINR_{n,m,r} = \frac{\beta_{n,m,r}g_{n,m}P_{n}^{t}}{\sum_{n'\in N, n'\neq n}\sum_{m'\in M_{n'}}\beta_{n',m',r}g_{n',m}P_{n'} + B_{r}N_{0}},$$
(2)

where  $\beta_{n,m,r}$  is a binary indicator to denote whether the  $r^{th}$  resource block of the  $n^{th}$  SBS is allocated to the  $m^{th}$  UE.  $g_{n,m}$  is the channel gain of the transmission link, which is decided by a free space propagation model.  $P_n^t$  denotes the transmission power of the  $n^{th}$  SBS and  $N_0$  denotes the noise power density.

We assume that UEs can support dual connectivity and can simultaneously connect to the MBS and SBS [17]. If the SBS is active, the UE will be served by the SBS. Otherwise, it will be served by the MBS. The energy-efficiency of the system can be defined as:

$$EE = \frac{\sum_{m \in M} b_m}{\sum_{n \in N} P_n + P_0},\tag{3}$$

where  $b_m$  denotes the throughput of the  $m^{th}$  UE, which is decided by both link capacity and arriving traffic.  $P_0$  and  $P_n$  denotes the power consumption of the MBS and the  $n^{th}$  SBS.

The optimization objective of the sleep control application is to achieve high energy-efficiency. Here we formulate the problem as:

m

s.

$$\max_{a_n} EE - \sum_{m \in M} \epsilon_m, \tag{4}$$

$$t. (1) - (3)$$

$$a_n \in \{0, 1, 2\}, \ \forall n \in \mathbb{N}$$

$$(4a)$$

$$\delta_n = \begin{cases} = 1, & if \ a_n = 0, \\ = 0, & else \end{cases}$$
(4b)

$$P_n = \begin{cases} P_w, & \text{if } a_n = 0, \\ 0.5P_w, & \text{if } a_n = 1, \\ 0.35P_w, & else \end{cases}$$
(4c)

where  $\epsilon_m$  denotes the packet drop rate of the  $m^{th}$  UE. A packet will be dropped if it exceeds the transmission delay constraint [18].  $a_n$  denotes the sleeping modes of the  $n^{th}$  SBS.  $a_n = 0$  indicates the SBS is in the active mode,  $a_n = 1$  indicates the SBS is in the sleep mode, and  $a_n = 2$  indicates the SBS is in deep sleep mode.  $P_w$  denotes the energy consumption of the SBS in active mode. The sleep mode can reduce energy consumption by 50%, and the deep sleep mode can reduce it by 65% [8].

To solve this problem, we use federated reinforcement learning (FRL) to promote privacy-preserving collaborative training. Each SBS holds a local deep reinforcement learning (DRL) model, which observes states and rewards from the environment and selects actions by choosing an adequate sleeping mode. The MBS serves as a global server in FRL, collecting local models from SBSs for model aggregation and distributing the global model as feedback. To attack the system, we suppose  $N^{mali}$  out of the N SBSs are malicious and can cause system performance degradation by updating malicious local models to the global server.

## IV. FEDERATED REINFORCEMENT LEARNING-BASED CELL SLEEP CONTROL

This section introduces the FRL-based cell sleep control application. Here DRL is applied in each SBS as a local model, and the optimal actions are selected by maximizing the longterm expected rewards.

The Markov decision process (MDP) of each local DRL is defined as follows:

• State: The state includes the sleeping mode of the SBS and the traffic load of the SBS and the MBS in the past 5 transmission time intervals which can be used to estimate the upcoming traffic load. It also includes the current delay and throughput of the SBS, which can be given as:

$$s_n = \{\delta_n, L_n, L_0, d_n, b_n\}, \forall n \in N,$$
(5)

where  $L_n$  denotes the traffic load of the  $n_{th}$  SBS.  $L_0$  denotes the traffic load of the MBS.  $d_n$  and  $b_n$  denote the delay and the throughput.

• Action: The action of sleep control is to choose an adequate sleeping mode for the SBS, which can be given as:

$$a_n = \{0, 1, 2\}, \forall n \in N,$$
(6)

• Reward: The reward function is defined as a combination of both quality of service (QoS) related indicators and the power consumption related cost, which can be given as:

$$R_n = \eta_1 b_n - \eta_2 \epsilon_n - \eta_3 P_n, \forall n \in N, \tag{7}$$

where  $\epsilon_n$  denotes the packet drop rate and  $b_n$  denotes the throughput.  $\eta_1$ ,  $\eta_2$  and  $\eta_3$  are the coefficients used to balance different rewards. When obtaining a high reward value, we expect the system to consume as little energy as possible while ensuring a high throughput. Therefore,



Fig. 2. Attack and defense models.

maximizing the given reward value is equivalent to maximizing the energy-efficiency and minimizing the packet drop rate.

On top of local models, we apply FRL to enable collaborative training and accelerate learning while keeping data locally and preserving privacy. In each FRL cycle, the local models will first perform local training according to local experience, which can be given as:

$$\theta_n^{t+1} = \theta_n^t + \alpha [r_n^t + \gamma \max_a Q(s_n^{t+1}, a; \theta_n^t) - Q(s_n^t, a_n^t; \theta_n^t)] \nabla Q(s_n^t, a_n^t; \theta_n^t),$$
(8)

where  $\theta_n$  denotes the local model parameters of the  $n_{th}$  SBS,  $\alpha$  denotes the learning rate and  $\gamma$  denotes the discount factor.  $Q(s_n^t, a_n^t; \theta_n^t)$  denotes the long-term expected reward of the  $n_{th}$  SBS choosing the action  $a_n^t$  under the state  $s_n^t$ . After local training, the local models are uploaded to the global server for model aggregation, which can be formulated as:

$$\theta_G^{t+1} = \sum_N^{n=1} w_n \theta_n^{t+1}$$
(9)

where  $\theta_G$  is the parameters of the global model.  $w_n$  is the weight of the  $n_{th}$  local model and it is decided by the number of training samples. In the scenario of FRL-based cell sleep control, we assume all the local models are equally weighted.

After the global model aggregation, the global model parameters are sent back to the SBSs and the local models are updated by replacing the local parameters with global parameters.

#### V. ATTACKS AND DEFENSE

This section presents the designed attack and defense models in the FRL-based cell sleep control scenario. Fig. 2 shows the structure of the investigated attacks and the proposed defense model. We proposed three attack models: free rider attacks, Byzantine data poisoning attacks, and backdoor attacks. We also proposed one defense scheme called refined-Krum.

## A. Attack models.

1) Free rider attacks.: Free riders refer to the FL participants who do not train their local models during the local training step [19]. As shown in Fig. 2, a benign BS will keep a memory buffer to store local experience and use it to train a benign local model. In contrast, a free rider does not train its local model and will submit the previously received global model as its own local model. The free rider is a passive attack method which means it will not actively destroy the global model or other local models. However, free riders are still malicious because they enjoy the resources and efforts of collaborators without contributing their own experience and training results during the collaboration. In this way, they can break the fairness of the FL system, and when the proportion of free riders increases, they will slow down the FL training.

2) Byzantine data poisoning attacks.: Byzantine data poisoning attacks are implemented by injecting poisoned data into the local memory buffer of the malicious participants and making participants train their local model in the wrong direction [10]. As shown in Fig. 2, the poisoned local model is trained with a mixture of local experience and poisoned data.

For our FRL model, we create poisoning data by giving the model a larger reward if the SBS chooses to be active and a smaller reward if the SBS chooses to sleep. The local model training in Eq. (8) on the poisoned data can be reformulated as:

$$\theta_n^{t+1} = \theta_n^t + \alpha [r_n^t + \sigma R^{ex} + \gamma \max_a Q(s_n^{t+1}, a; \theta_n^t) - Q(s_n^t, a_n^t; \theta_n^t)] \nabla Q(s_n^t, a_n^t; \theta_n^t),$$
(10)

where  $R_{ex}$  denotes the extra reward and  $\sigma$  is an indicator to denote whether to add an extra reward or to reduce the reward. If  $a_n^t = 0$ ,  $\sigma = -1$ . Otherwise,  $\sigma = 1$ .

After local training, the poisoned local model will be uploaded to the global server and the effect of poisoned data will then be propagated to other local models during the global model aggregation.

3) Backdoor attacks.: Different from regular data poisoning attacks, backdoor attacks define backdoor tasks to make a separation from the main tasks. Main tasks refer to the task defined by the original training data. Backdoor tasks are defined by the attacker, and they represent a special input data pattern that rarely appears in the original training data. In FRL, the backdoor tasks can be defined by specific states. Every time the backdoor attackers want to attack the system model, they can trigger the backdoor pattern and make the model select abnormal actions.

In our sleep control scenario, we define the backdoor task as an SBS with a very high traffic load. It can be easily triggered by putting a UE with an extremely heavy traffic load into the system, and the trigger result is defined as the SBS will turn into deep sleep mode. As shown in 2, the backdoor-ed local model is trained with the mixture of local experience data and synthetic backdoor task training data.

Benefiting from separating the main and backdoor tasks, the backdoor attacks are stealthy and difficult to detect. Since the backdoor pattern is defined by the attacker and unknown to others, the model owner will only test it with main tasks while doing a security check. In this way, they could not find the model had already been backdoor-ed.

# B. Defense model.

We propose a refined-Krum defense model based on the existing secure FL aggregation method Krum [9]. As shown in Fig. 2, the refined-Krum is deployed at the global server and will be performed before global aggregation during each FL iteration. In this subsection, we first introduce the Krum defense scheme and then illustrate how the refined-Krum model is defined.

1) Krum defense: Krum is proposed in [9] and its core idea is to assume that all benign local models are similar. Therefore, the malicious models can be found by measuring the similarity of all the local models by the Krum distance.

In the Krum defense, the Krum distance for each local model is first calculated. In the first step, the Euclidean distance between parameters of the  $n^{th}$  local model and the global model in the last FL iteration is calculated as:

$$G_n^{t+1} = \left\| \theta_n^{t+1} - \theta_G^t \right\|_2 \tag{11}$$

Then, the distance between the  $n^{th}$  local model and  $k^{th}$  local model can be given as:

$$D_{nk}^{t+1} = \left\| G_n^{t+1} - G_k^{t+1} \right\|_2 \tag{12}$$

The distance between each local model and all other local models is then added. In this way, the Krum distance for each local model can be obtained, which can be given as:

$$D_n^{t+1} = \sum_{k \in N} D_{nk}^{t+1}$$
(13)

Finally, the Krum defense will select the local model with the smallest Krum distance and replace the global model with the selected local model.

2) Refined-Krum: Although the Krum defense scheme is proven to be effective in some cases, choosing only a local model for global aggregation is quite unstable and it may not get the full benefit of FL. Therefore, we designed a new defense algorithm called refined-Krum. It can be concluded into four steps, which are calculating the similarity gaps, estimating the number of malicious participants, identifying malicious participants and secure aggregation.

- Calculating the similarity gaps. In the first step of refined-Krum, we calculate the Krum distance of each local model to evaluate the similarities between models. But instead of only selecting the local model with the smallest Krum distance, we sort all the models by their Krum distances from the smallest to the largest and calculate the gap between two adjacent Krum distances.
- Estimating the number of malicious participants. With the gap values calculated in the first step, we can then estimate the number of malicious participants by finding the maximum gap between the given distance list. This is based on the assumption that most models are benign and the malicious models are quite different from the benign ones. So there will be a large gap between the similarities.

- Identifying malicious participants. If the maximum gap is much larger than the average, we suppose it could precisely separate malicious models from benign ones and decide the threshold for the Krum distance. If the Krum distance of the  $n^{th}$  local model is larger than the threshold value, the  $n^{th}$  SBS is treated as a malicious participant. Other SBSs are treated as benign participants. On the other hand, if the maximum gap is close to the average gap, we assume that the threshold cannot be accurately determined and to mitigate the risk of being attacked, only one benign model will be selected.
- Secure aggregation. After identifying the malicious participants, we can perform secure aggregation by admitting only benign models into the global model aggregation. This prevents the malicious data from influencing the global model and the benign SBSs. At the same time, we will make MBS take over the sleep control for the malicious SBSs and prevent attackers from controlling these SBSs.

#### VI. NUMERIC RESULTS

### A. Simulation settings.

In the simulation, we consider 8 SBSs, and each SBS has 10 UEs. The fixed power consumption of MBS and SBSs are 40W and 20W, respectively [17]. The radius of MBS and SBSs are 400m and 100m, respectively. The available bandwidth for each SBS is 20 MHz, and for MBS is 10 MHz.  $\eta_1$ ,  $\eta_1$  and  $\eta_1$  are respectively 0.1, 1 and 0.01. During the simulation, we change the average traffic load of each SBS from 30 Mbps to 70 Mbps and compare the system energy-efficiency under different attack and defense models. We simulate a 24-hour typical residential area traffic pattern in each TTI, which is given from [20].

For the free rider attacks, we include two free riders in the network. For Byzantine data poisoning attacks and backdoor attacks, we have one malicious SBS in the networks. We assume the proportion of poisoned data or backdoor task training data of malicious SBSs is 5%. Therefore, the proportion of poisoned data in the total data of all the SBSs is 0.625%.

#### B. Simulation results

Firstly, we compare the simulation results of FL and independent learning-based cell sleep control in a fully secure environment. Independent learning (IL) means there are no collaborations between SBSs, and each SBS will train a DRL model according to their local buffer data. Fig. 3 shows the convergence curves of FL and independent learning when the average traffic load of each SBS is 40 Mbps and in each TTI we run 24-hour traffic. The FL algorithm we use during the simulations is FedAvg. The FL performs much better than IL and has higher rewards, which demonstrates the effectiveness of the FL algorithm.

Then we add three different attacks to the FRL-based sleep control scenario. The system energy-efficiency under free rider attacks, data poisoning attacks, and backdoor attacks are shown in Fig. 4. The system performance in a secure



Fig. 3. The convergence curves of FL and independent learning.



Fig. 4. The system energy-efficiency under different attacks.

environment with no attacks is also compared. It can be observed that three kinds of attacks can degrade the system performance to different levels. Backdoor attacks can be seen as the most effective attacker. When the average traffic load is 70 Mbps, the backdoor attacks can reduce the system energyefficiency by 52%. Among the remaining two attacks, the data poisoning attacks are more effective than the free rider attacks, even with fewer attackers. From this observation, we can also conclude that while designing defense mechanisms for FRL, it is more important to prevent malicious participants from being involved in the aggregation than to ensure that the benign participants are involved in the aggregation. When the average traffic load is 70 Mbps, a malicious SBS with poisoning attacks can reduce the system energy-efficiency by 18%.

When it comes to defense, the system energy-efficiency under our proposed refined-Krum defense scheme is shown in Fig. 5. We only defend against poisoning and backdoor attacks because even if we can detect free riders, we cannot force them to contribute to the model. As it can be observed, for data poisoning attacks, the system energy-efficiency after the defense is very close to the situations with no attack. We can conclude that the defense can almost fully recover the system from data poisoning attacks with a limited number of attackers. The defense scheme can also significantly improve system performance and increase energy-efficiency for



Fig. 5. The system energy-efficiency under the proposed defense scheme.



Fig. 6. The system energy-efficiency under the Krum defense scheme and the proposed defense scheme.

backdoor attacks. However, the energy-efficiency defense is still lower than the performance in a secure environment. This indicates that our proposed defense scheme is quite effective for some attacks but less effective for others.

In Fig. 6, we further compare our proposed refined-Krum defense scheme with the existing Krum defense scheme. For both kinds of attacks, our proposed refined-Krum defense scheme can get a higher energy-efficiency compared with the Krum defense scheme. Also, refined-Krum is more stable with a smaller confidence interval.

#### VII. CONCLUSION

In this work, we studied how to attack a FRL-based cell sleep control scenario in a wireless network. We considered three types of attacks that could perform on wireless networks, which are free riders, Byzantine data poisoning attacks and backdoor attacks. According to the simulation results, these attacks can degrade system performance with lower energyefficiency. We also proposed a defense scheme called refined-Krum to defend against these attacks. The simulation results show that our proposed defense scheme can effectively increase the system energy-efficiency and prevent the system from attacks. In our future research, we plan to investigate more advanced attacks and improved defense schemes.

# ACKNOWLEDGEMENT

This work has been supported by MITACS and Ericsson Canada, and NSERC Collaborative Research and Training

### Experience Program (CREATE) under Grant 497981.

#### References

- M. Elsayed and M. Erol-Kantarci, "AI-enabled future wireless networks: challenges, opportunities, and open issues," *IEEE Vehicular Technology Magazine*, vol. 14, no. 3, pp. 70–77, Sep. 2019.
- M.Elsayed and M.Erol-Kantarci, "Al-enabled radio resource allocation in 5G for URLLC and eMBB users," *IEEE 5G World Forum, 5GWF 2019* - *Conference Proceedings*, pp. 590–595, Sep. 2019.
- [3] H. Zhou, M. Erol-Kantarci, and H. V. Poor, "Knowledge Transfer and Reuse: A Case Study of AI-enabled Resource Management in RAN Slicing," *IEEE Wireless Communications*, pp. 1-10, Nov. 2022.
- [4] H. Zhang, H. Zhou, and M. Erol-Kantarci, "Federated Deep Reinforcement Learning for Resource Allocation in O-RAN Slicing," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, pp. 958-963, Dec. 2022.
- [5] V. Mothukuri, R. M. Parizi, S. Pouriyeh, Y. Huang, A. Dehghantanha, and G. Srivastava, "A survey on security and privacy of federated learning," *Future Gener. Comput. Syst.*, vol. 115, pp. 619–640, Feb. 2021.
- [6] L. Lyu, H. Yu, X. Ma, C. Chen, L. Sun, J. Zhao, Q. Yang, and S. Philip., "Privacy and robustness in federated learning: Attacks and defenses," in *IEEE transactions on neural networks and learning systems*, pp. 1-21, Nov. 2022.
- [7] M. Usama and M. Erol-Kantarci, "A survey on recent trends and open issues in energy efficiency of 5G," *Sensors*, vol. 19, no. 14, p. 3126, Jul. 2019.
- [8] C. Liu, B. Natarajan, H. Xia, "Small cell base station sleep strategies for energy efficiency". in *IEEE Trans. Veh. Technol.* pp. 1652–1661, Mar. 2015.
- [9] P. Blanchard, E. M. E. Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems, pp. 119–129, Dec. 2017.
- [10] V. Tolpegin, S. Truex, M. E. Gursoy, and L. Liu, "Data poisoning attacks against federated learning systems," arXiv preprint arXiv:2007.08432, Sep. 2020.
- [11] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," arXiv preprint arXiv:1807.00459, Apr. 2018.
- [12] A. Anwar and A. Raychowdhury, "Multi-task federated reinforcement learning with adversaries," *CoRR*, vol. abs/2103.06473, Mar. 2021.
- [13] E. Ma and R. Etesami, "Local Environment Poisoning Attacks on Federated Reinforcement Learning," arXiv preprint arXiv:2303.02725, Apr. 2023.
- [14] T. Zheng and B. Li, "Poisoning attacks on deep learning based wireless traffic prediction," in *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications*, pp. 660–669, May. 2022.
- [15] Y. Shi and Y.E. Sagduyu, "How to Launch Jamming Attacks on Federated Learning in NextG Wireless Networks," in *IEEE Globecom Workshop on 5G and Beyond Wireless Security (Wireless-Sec)*, pp. 945-950, Jan. 2022.
- [16] M. Masoud, M.G. Khafegy, E. Soroush, and D. Giacomelli, "Reinforcement Learning for Traffic-Adaptive Sleep Mode Management in 5G Networks," *IEEE Annual International Conference*, pp. 1-6, Aug. 2020.
- [17] M. A. Habib, H. Zhou, P. E. Iturria-Rivera, M. Elsayed, M. Bavand, R. Gaigalas, S. Furr, and M. Erol-Kantarci, "Traffic Steering for 5G Multi-Rat Deployments using Deep Reinforcement Learning," in IEEE Consumer Communications and Networking Conference (CCNC), pp. 164-169, Jan. 2023.
- [18] A. E. Amine, P. Dini, and L. Nuaymi, "Reinforcement learning for delayconstrained energy-aware small cells with multi-sleeping control," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, pp. 1–6, Jun. 2020.
- [19] Y. Fraboni, R. Vidal, and M. Lorenzi, "Free-rider attacks on model aggregation in federated learning," in *Proc. Int. Conf. Artif. Intell. Statist.*, pp. 1846–1854, Jun. 2021.
- [20] H. Zhou, L. Kong, M. Elsayed, M. Bavand, R. Gaigalas, S. Furr, and M. Erol-Kantarci, "Hierarchical reinforcement learning for RISassisted energy-efficient RAN," in *Proc. IEEE Global Commun. Conf.* (GLOBECOM), pp. 3326–3331, Dec. 2022.