Dang, Yongchao; Karakoc, Alp; Jäntti, Riku

Graphic Neural Network based GPS Spoofing Detection for Cellular-Connected UAV swarm

Published in:
2023 IEEE 97th Vehicular Technology Conference, VTC 2023-Spring - Proceedings

DOI:
10.1109/VTC2023-Spring57618.2023.10200557

Published: 23/06/2023

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Graphic Neural Network based GPS Spoofing Detection for Cellular-Connected UAV swarm

Yongchao Dang *, Alp Karakoc * and Riku Jäntti *
* Aalto University, Espoo, Finland
* Email: firstname.lastname@aalto.fi.

***Abstract*—The cellular-connected Unmanned Aerial Vehicles (UAVs) are emerging as integral components of the 5G and beyond system due to their mobility and flexibility. Compared to a traditional single UAV, a flock of UAVs established as a UAV swarm can implement diverse distributed applications economically and efficiently, such as cooperatively smart agriculture, joint search and rescue, and supplementing temporary network connections. However, the GPS spoofing attack can manipulate UAV swarm locations and distort UAV swarm topology, which threatens the security of swarm communication and control. This paper proposes a Graphic Neural Networks (GNN) based GPS spoofing detection approach for cellular-connected UAV swarms. Especially, we propose a system in which the GNN model is used to detect GPS spoofing attacks by analyzing the similarity between the swarm GPS topology and communications topology. To evaluate the proposed neural networks, we use a bipartite graph and Hungarian algorithm to build a UAV swarm simulator. The results show that GNN can efficiently compute topologies' similarity and detect GPS spoofing attacks. For instance, for a UAV swarm consisting of 10 UAVs, GNN detects the spoofing with accuracy over 90% and computation time of fewer than 10 milliseconds using Intel Core 1.6 GHz processor.**

***Index Terms*—Unmanned Aerial Vehicles (UAVs), UAV swarm, GPS spoofing detection, and Graphic Neural Networks (GNN).**

I. Introduction

The cellular-connected Unmanned Aerial Vehicles (UAVs) are emerging as imperative parts of the upcoming 5G and beyond the system. Specifically, A flock of UAVs, or named UAV swarm, has been envisioned in different kinds of applications, such as cooperative rescue, search, and network recovery economically and efficiently [1]. A safe and reliable navigation system is compulsory for cellular-connected UAV swarms to complete the mission successfully. For example, the safe navigation system can help the swarm members avoid collisions and optimize the trajectory as well as support the swarm formation flocking and coordinated path planning [2].

The Global Navigation Satellite System (GNSS), specifically GPS, is adopted by cellular-connected UAVs because of its global coverage and accuracy. However, civil GPS signals are unencrypted and vulnerable to signal spoofing attacks [3]. In practice, the GPS spoofer can use Software Define Radio (SDR) to generate fake GPS signals and information, which cause the GPS receiver to compute wrong positions [3]. Indeed, the GPS spoofer can manipulate the swarm topologies and deviate the swarm from its optimized trajectories [2]. In addition, the spoofed GPS positions also have an impact on the swarm communication and control, because the swarm Ad-hoc On-demand Distance Vector (AODV) routing protocol lacks strong authentication on the node and is vulnerable to the spoofing attack [3]. Thus, it is necessary to endow the swarm with the ability to verify the GPS navigation information and withstand GPS spoofing attacks.

Several countermeasures have been proposed to protect UAVs against GPS spoofing attacks, including GPS navigation signals analysis (e.g., [4]–[6]) and GPS navigation message authentication (e.g., [7]–[10]). The GPS signal analysis approaches to detect the GPS spoofing attack by estimating and comparing the radio fingerprinting of GPS satellites [4], such as the Direction of Arrival (DoA) of received GPS signals [5] or the Time of Arrival (ToA) of the fake GPS signals [6]. In addition to GPS signal analysis, GPS navigation message authentication methods counteract GPS spoofing by generating cryptographically signed navigation messages [7]. Wu *et al.* inserted the digital signature into the navigation message with the help of the elliptic curve digital signature algorithm [8]. Furthermore, the authors in [9] used SM cryptographic algorithms to protect navigation messages from modification for BeiDou II. Furthermore, Nicola *et al.* in [10] evaluated the authentication of open service navigation messages for the Galileo navigation system based on the efficient timed stream loss tolerant authentication protocol.

Although the above-mentioned approaches have shown effectiveness against GPS spoofing attacks, they require a lot of processor resources and have cost energy consumption at the GPS receiver, which limits their implementation in real UAV system that has limited computation power and battery capability [11]. In addition to GPS navigation signals analysis and GPS navigation message authentication approaches, the UAV Inertial Navigation System (INS) is used to detect GPS spoofing by comparing the difference between the position inferred from the Inertial Measurement Unit (IMU) and the one reported by the GPS receiver [12]. If the difference is above a preset threshold value, the GPS position is spoofed. A key advantage of INS is its resilience to spoofing attacks as it does not rely on any external signal. However, the main weakness of INS is the error accumulation of the IMU measurements over time, which can seriously affect the detection accuracy.

The Mobile Positioning System (MPS) is another available GPS spoofing detection method for UAVs and UAV swarms. To enhance Long Term Evolution (LTE) support for Unmanned Aerial Systems (UAS), the 3rd Generation Partnership Project (3GPP) had defined new standards that allow the UAS to access MPS services to locate and track the UAVs. In this

vein, Dang *et al.* in [13] proposed the Adaptive Trustable Residence Area (ATRA) to cross-check the validity of GPS information reported by UAV. However, the ATRA method requires at least three base stations (BSs) at the same time. To overcome these weaknesses, Dang *et al.* in [14] introduced the deep learning method into the edge server, which allows detecting the GPS spoofing for cellular-connected UAVs by using one BS. Furthermore, the authors in [15] used transfer learning methods in edge servers to train the model for GPS spoofing detection, where the use of transfer learning can increase detection accuracy and decrease detection latency.

Despite the fact that the proposed deep learning and transfer learning approaches demonstrate effectiveness in detecting GPS spoofing, they are designed for one single cellular-connected UAV and do not consider the case of the cellular-connected UAV swarm. In fact, the GPS attacker can manipulate the swarm formation by spoofing a specific UAV member position, which may lead AODV to become unstable, increase communication latency and decrease collaboration in the swarm [2]. Additionally, the above approaches support GPS spoofing detection for cellular-connected UAVs by running multiple detection procedures on the edge server, one per UAV, which may lead to congestion in the detection system when a large UAV swarm connects with one base station. In this paper, we propose a GNN-based GPS spoofing detection approach for the cellular-connected UAV swarm, where GNN uses Graphic Conventional Network (GCN) layers and recurrent network layers to detect GPS spoofing attacks on the swarm by comparing the difference between the swarm GPS topology information of and the communication topology. The following are the major contributions of this paper.

- We investigated a mobile cellular network-assisted Flying Ad-hoc Network (FANET) system (see Fig. 1), where the UAV swarm leader is connected with a base station and controlled by the edge UAVs Flight Controller (UFCs). In addition, the UFCs can monitor and verify GPS positions reported by swarm members through Air-to-Air (A2A) and Air-to-Ground (A2G) communication links.
- To detect the GPS spoofing against the swarm, we designed a GNN to the edge UFCs in order to analyze the swarm-reported GPS information and the communication links information. Specifically, the GCN layers are used to compute the spatial similarity between the swarm GPS topology and communications topology while the recurrent networks are for acquiring the temporal similarities of the swarm topology changes caused by GPS spoofing attacks.
- To evaluate the proposed neural networks, we use a bipartite graph and Hungarian algorithm to build a UAV swarm trajectory simulator that demonstrates the impacts of GPS spoofing attacks on the swarm topology and provides the data for GNN training and evaluation.

The remainder of this paper is organized as follows. Section II introduces the system model and problem formulation. Section III provides details on the proposed GNN-based GPS spoofing detection method. The results are given in Section IV. Conclusion and future work are presented in Section V.

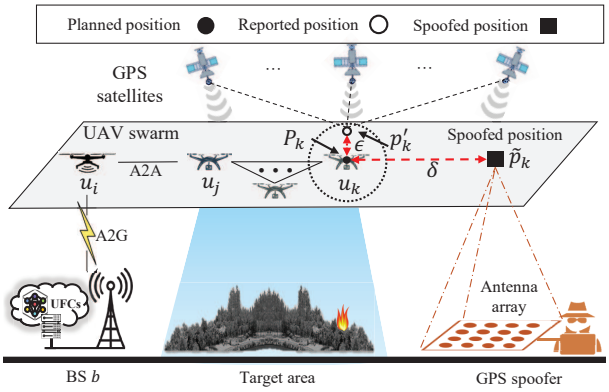


Fig. 1. Cellular-connected FANET system.

II. System Model and Problem Formulation.

A. System Model

1) Network Model: As illustrated in Fig.1, we consider an edge network scenario consisting of one BS b, a UAV swarm containing N UAVs, and a GPS spoofer equipped with an antenna array. The BS connects with the edge server that runs the UFCs to control the swarm. The aerial UAVs swarm is organized in a FANET manner and uses the AODV routing protocol for communication and collaboration among the swarm leader u_i, the swarm members u_j and u_k, $0 < i, j, k \leq N$ and $i \neq j \neq k$. Specifically, the leader of the swarm u_i is the gateway between the UFCs and other members of the swarm, which can expand the range of communication of the swarm while reducing spectrum interference. In addition to the UAV swarm and BS, there is a GPS spoofer with an antenna array to spoof the GPS signal, which can cause the swarm to deviate from the planned position and arise collision risks in the swarm.

2) Channel Model: The A2A channel between u_i and u_j follows the free-space path loss model, and the A2G channel model between u_i and b is defined in [16] by the 3GPP.

3) Attack Model: Let (x_b, y_b, h_b) denote the location of the BS position, and (x_k, y_k, h_k) represents u_k position. Casting aside GPS spoofing and GPS error, u_k should be at the planned waypoint p_k at time t. In the presence of a GPS error, the report position $p_k^{'}$ is close to p_k with an error ϵ. Once GPS spoofed, the UAV u_k locates at $\tilde{p}_k$ that deviates from p_k with δ, where $\epsilon \leq dE < \delta$ and dE is the system's tolerable GPS margin error (see Fig.1). The attacker can either spoof a specific UAV GPS position or broadcast fake GPS signals into the air and spoof all the UAVs' positions. The first type of spoofing can manipulate some UAVs' positions and change the swarm formation, while the second type of spoofing leads the whole swarm to a wrong destination without any changes to the swarm formation. We assume that the spoofing attacks will not affect swarm and BS communication.

B. Problem Formulation

The cellular-connected UAV swarm system is modeled as an undirected graph $\mathcal{G}$, $\mathcal{G} = \{V, E, W\}$. V is the node-

set, and $V = \{v_0, ..., v_i, ..., v_N\}$, where v_0 represents the BS b and node v_i denotes the UAV u_i. E is the edge set that represents the interaction between BS and the swarm, $E \subseteq \{(u_i, u_j) : u_i, u_j \in V, 0 \leq i \neq j \leq N\}$ particularly. The weighted adjacency matrix W is the interaction strength, $W = [w_{ij}] \in \mathbb{R}^{(N+1)\times(N+1)}(w_{ij} \geq 0)$. The interaction strength for the cellular-connected UAV swarm can be extracted from GPS positions or generated from wireless connections. Let $\mathcal{G}_g$ denote the GPS topology and $\mathcal{G}_l$ represent the communication topology for the cellular-connected UAV swarm. The interaction strength w_{ij} for $\mathcal{G}_g$ stands for the distance d_{ij}, $w_{ij} = 0$ if $d_{ij} > r_{ij}$, where r_{ij} is the max communication range between node i and node j. Correspondingly, the interaction strength w_{ij} for $\mathcal{G}_l$ represents the path loss value from node i to node j, $w_{ij} = \overline{L}_{ij}$ when $i = 0$ or $j = 0$, otherwise $w_{ij} = L_{ij}$.

Theoretically, the path loss increases with increasing distance between the transmitter and receiver. In such a dynamic scenario, the strength of GPS interaction is close to the communication path loss interaction in the cellular-connected UAV swarm at a given time. However, fake or false GPS positions from GPS spoofing correspond to the deviation between the swarm communication topology and the GPS location topology. Therefore, the similarity between $\mathcal{G}_l$ and $\mathcal{G}_g$ can indicate GPS spoofing in the cellular-connected UAV swarm. Let $\mathfrak{S}_{gl}$ denote the similarity between $\mathcal{G}_l$ and $\mathcal{G}_g$. Hence, the swarm GPS spoofing detection problem can be formulated as a threshold-based hypothesis test, seen in (1).

$$\begin{cases} H_0: & \mathfrak{S}_{gl} > \tau, \\ H_1: & \mathfrak{S}_{gl} \leq \tau, \end{cases} \tag{1}$$

where τ denotes a threshold of hypothesis testing and H_0 is the null hypothesis that indicates GPS spoofing while H_1 stands for no GPS spoofing. H_0 is accepted if $\mathfrak{S}_{gl}$ is above τ. Otherwise, H_1 is accepted. Noteworthy, $\mathfrak{S}_{gl}$ can be represented by Graph Edit Distance (GED) [17].

Although there are algorithms for GED computation, all of them cannot manage graphs with more than 16 nodes in a reasonable time [17]. In addition, the swarm communication topology is influenced by environmental factors, such as clouds, temperature, and vapor, which may lead to a wrong decision on GPS spoofing. Thereby, the threshold-based hypothesis testing in (1) faces the following challenges. First, the current GED computations are high time consumption that can lead to fatal damage to the swarm due to detection latency. Secondly, the swarm communication topology is more likely to be affected by the environment, which may increase spoofing detection errors. Thirdly, the threshold setting has an immediate impact on the accuracy of the hypothesis testing, i.e. a bigger threshold setting leads to a higher probability of miss detection while a smaller threshold setting results in a higher probability of false alarms.

To address the aforementioned challenges, we employ GNN model on the edge server to devise an effective GPS spoofing detection approach for the cellular-connected UAV swarm, seeing in the following section.

III. GNN BASED GPS SPOOFING DETECTION

Fig.2 shows GNN architecture that includes GCN layers, attention and tensor network layers, recurrent network layers, and dense layers. Specifically, GCNs, attention, and tensor networks have the same architecture as the SimGNN in [18], which are in charge of topological spatial similarity measurements while recurrent network layers and dense layers are responses to extract topological temporal changes.

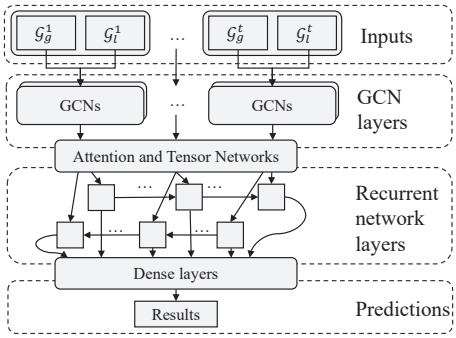


Fig. 2. GNN architecture.

The GCN layers uses the GPS topology $\mathcal{G}_g$ and the communication topology $\mathcal{G}_l$ as inputs and produces the similarity score *Similarity*$(\mathcal{G}_g, \mathcal{G}_l)$. Firstly, the Graph Convolutional Networks (GCNs) embed the swarm GPS topology and one of its nodes into Euclidean vector space. Precisely, all N nodes in a graph $\mathcal{G}$ have N node embeddings [18]. Following, the attention mechanism emphasizes each node's importance and similarity within one graph embedding for graph-level embeddings. However, the attention emphasized that graph embeddings lose the small substructures features, which debilitates the authenticity of the final similarity [18]. To solve this problem, both Neural Tensor Network (NTN) and pairwise node comparison are employed for analyzing graph-level embeddings in order to extract fine-grained node-level information [18]. More specifically, the histogram features of node-level embeddings are used in the Pairwise Node Comparison (PNC) to represent node-level features [18]. At the end, fully connected layers are employed to aggregate node-level and graph-level features to produce a similarity score, $\mathfrak{S}_{gl}$, $\mathfrak{S}_{gl} =$ *Similarity*$(\mathcal{G}_g, \mathcal{G}_l)$.

The GPS spoofing attack is a continuous procedure that manipulates the swarm and its member positions constantly. It is difficult for the GCN layers to capture the topological changes of the swarm over time. Fortunately, the recurrent network layers and dense layers are good at analyzing the time sequence data and detecting abnormal topology changes from GPS spoofing attacks. Initially, the unit of recurrent network layers has two kinds of inputs, one input for the recent past graph similarities and another input for the present similarity, which allows for making a decision in a combination with graph temporal similarities [19]. However, the traditional recurrent network unit has an impediment to long-term prediction, which is caused by the vanishing or explosion of the gradient in adaptive neural networks [19]. Two variants of the recurrent networks have been developed, namely, the

long-short-term memory (LSTM) model and the gated recurrent units (GRUs) model, based on a gated mechanism to solve the above problems [19]. The LSTM uses three gates, input, output, and forget gate, to extract the temporal features from time sequences, while the GRU has two gates with two activation functions for combining the past and present features. The evaluation of the GNN model is illustrated in Section IV.

IV. Performance Evaluation

A. Simulation Setting

We develop a swarm simulator using Python 3.6 and Tensorflow 2.1.0 to evaluate the performance of the proposed spoofing detection algorithms. In this simulator, Python is used to set up the simulation platform, and Tensorflow is employed to build the GNN and RNN models.

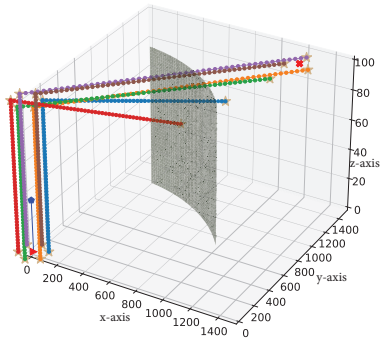


Fig. 3. Simulation platform with 6 UAVs in the swarm.

Fig.3 shows the simulation platform that contains one base station and an aerial UAV swarm distributed in a 3D $1400 \times 1400 \times 100\ m^3$ space, where the base station locates at the origin point while the swarm starts at the initial waypoints and towards to the target waypoints. In particular, the spherical surface is the boundary of the base station coverage. It is worth mentioning that the swarm needs to arrive at a given altitude and then move to their next waypoints, and then the bipartite graph and Hungarian algorithms are used to optimize the swarm trajectory from the given altitude positions to target positions. The simulation platform settings are illustrated in Table I, where the target positions of the swarm are two different locations, $\mathcal{P}_1$ denote the normal end position, while $\mathcal{P}_2$ stands for the spoofed end position, $\mathcal{P}_1, \mathcal{P}_2 \in \{(x, y, z)|x, y \in \{1200, 1250, ..., 1300\}, z = 100\}$. In addition, $\mathcal{N}$ is the Gaussian noise on the swarm positions caused by random winds. dE is preset as $\{10, 15, 20, 25, 30\}$ to simulate the GPS error tolerance of different scenarios. The channel frequency is set at 5.0 GHz for A2A communication and 2.4 Ghz for A2G communication.

The GNN has three GCNs layers with $(128, 64, 32)$ filters for embedding graph nodes, one NTN layer with 16 neurons for extracting nodes features, and one PCN with 16 neurons and 32 histogram bins for preserving link features, and recurrent network layers consist of one bidirectional layer (LSTM or GRU unit) followed by a dense layer and one output layer. The LSTM/GRU layer and the dense layer have the same number of neurons chosen from $\{10, 15, ..., 40\}$, while the output layer has one single neuron to make the final spoofing decision.

TABLE I
Parameter Settings of the Simulation Platform.

Parameter	Definitions	Value(s)
$\mathcal{P}_S$	Swarm initial position	(0,0,0)
$\mathcal{P}_T$	Swarm target positions	($\mathcal{P}_1$,$\mathcal{P}_2$)
$\mathcal{P}_B$	Base station position	(0,0,0)
N	Swarm UAVs' number	6,8,10
r_S	Swarm range size	100 m
r_U	UAV communication range	300 m
r_B	BS communication range	1200 m
$\mathcal{N}$	Random wind noise	$\mathcal{N}(0, 5)$
f_a	A2A channel frequency	5.0 GHz
f_g	A2G channel frequency	2.4 GHz
dE	GPS error	10,15,...,30 m

Since GCNs, attention, and tensor networks are applied for topological spatial similarity measurements while recurrent network layers and dense layers are responses to extract topological temporal changes, they are trained separately in order to reduce the model training time as well as prevent overfitting. The GNN has been trained for 200 epochs on 100 graphs and tested with 100 graphs. In particular, the Adam optimizer is used during GNN training with a learning rate of 0.001 and a weight decay of 0.001 for GCN layers while a learning rate of 0.0001 for recurrent network layers. Supplementally, the weight decay is also called L2 regularization and is used to accelerate training processes as well as prevent model overfitting. In addition, the dropout is also employed after GCNs layers with a rate of 0.9 to reduce overfitting and improve generalization. Moreover, the activation function $sigmoid$ is applied in the dense layer and the output layer to produce the final spoofing probability.

B. Performance Metrics

1) Normalized Graph Edit Distance (NGED): NGED is used to transform the graph edit distance into the graph similarity score. In [20], NGED is defined as:

$$NGED(\mathcal{G}_1, \mathcal{G}_2) = Similarity(\mathcal{G}_1, \mathcal{G}_2) = e^{-\frac{\text{GED}(\mathcal{G}_1, \mathcal{G}_2)}{(|\mathcal{G}_1|+|\mathcal{G}_2|)/2}}, \quad (2)$$

$Similarity(\mathcal{G}_1, \mathcal{G}_2)$ is in range of $(0, 1]$ and there is a one-to-one mapping between GED$(\mathcal{G}_1, \mathcal{G}_2)$ and $Similarity(\mathcal{G}_1, \mathcal{G}_2)$, conspicuously. NGED is used in SimGNN inputs and output during the model training and evaluation processes.

2) Mean Squared Error (MSE): MSE is the loss function for evaluating the performance of the GCN networks, which is expressed as

$$\mathcal{L} = \frac{1}{T}\sum_{t=1}^{T}(\mathfrak{S}_{gl}^t - \hat{\mathfrak{S}}_{gl}^t)^2, \quad (3)$$

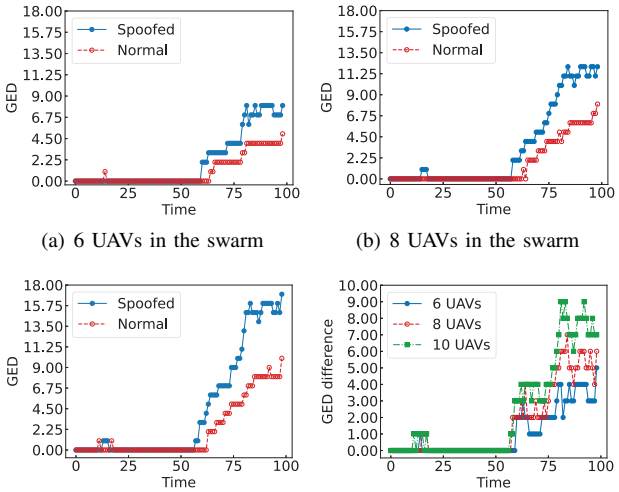
where $\mathcal{L}$ is MSE and $\mathfrak{S}_{gl}^t$ is the graph similarity score at time t while $\hat{\mathfrak{S}}_{gl}^t$ is the graph similarity prediction at time t.

3) The Binary Cross-entropy Loss (BCL): BCL is employed to assess the performance of the recurrent network layers, which is expressed as

$$\mathcal{B} = -\sum_{i=1}^{n} \hat{y}_i \log y_i + (1 - \hat{y}_i) \log(1 - y_i), \tag{4}$$

where $\mathcal{B}$ represent BCL and $\hat{y}_i$ is the i^{th} predicted spoofing probability while y_i is the i^{th} ground truth label, $y_i = 1$ representing no spoofing while $y_i = 0$ denoting GPS spoofed. $y_i = 0$ if and only if the average deviation of the swarm is outside the system GPS error tolerance.

C. Performance Results



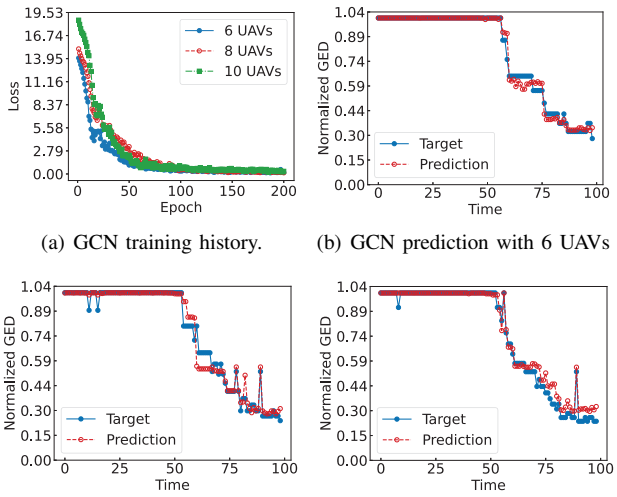
(a) 6 UAVs in the swarm (b) 8 UAVs in the swarm

(c) 10 UAVs in the swarm (d) GED difference.

Fig. 4. GPS spoofing attack on UAV swarms with $P_1 = (1200, 1200, 100)$ (Normal) and $P_2 = (1300, 1300, 100)$ (Spoofed). The spoofing starts at around time 60.

1) The swarm simulation platform data evaluation: Fig.4 illustrates the GED of different UAV swarms with GPS spoofing attack, where the normal position locates at $P_1 = (1200, 1200, 100)$ while the spoofed position is relocated to $P_2 = (1300, 1300, 100)$.

It can be observed from Fig.4 that the GPS spoofing attack has more impact on a bigger swarm. The reason is that the GPS spoofing attack can result in more changes to a bigger swarm topology than a small swarm. It is easy to see in Fig.4 that there are some flaws in the normal GED, which actually come from random winds, GPS errors, or communication noise. In practice, the random wind can cause the UAV to hang near to its planned GPS positions and bad weather can increase GPS signal error and decrease GPS accuracy. In fact, both random wind and GPS errors can result in the accumulation of the swarm GPS location topology errors. Additionally, swarm communication is also influenced by environmental conditions that can make swarm communication unstable. Unstable communication will be reflected in the path loss values and will further impact the swarm communication topology. So, it is reasonable that the GED between the swarm GPS topology and communication is different without GPS spoofing.

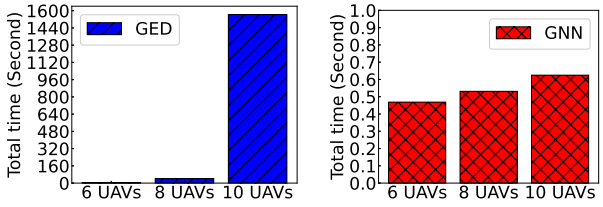


(a) GCN training history. (b) GCN prediction with 6 UAVs

(c) GCN prediction with 8 UAVs (d) GCN prediction with 10 UAVs

Fig. 5. GCN training history and prediction.

2) GNN based graph similarity prediction approach: Fig.5(a) depicts the GCN layers training history, where the MSE defined in (2) is used as a loss function on 6, 8, and 10 UAV swarm data sets. The training history shows that GCN layers can fit all swarm data sets with a small loss value near zero, which means the trained GCN layers can predict well on the training data set. It can also be observed from the training history that GCN layers are trained faster on the swarm with a smaller size. The reason behind the phenomenon is that there are fewer neurons working in the GCN layers when the swarm size is small. The performance of GCN layers on the test data sets are shown respectively in Fig.5.(b-d). It is clear that the trained GCN layers can also fit well on the test data set as well as keep the normalized GED trending. Thus, the similarity between GPS topology and communication topology can be used to detect GPS spoofing attacks on cellular-connected aerial swarms.



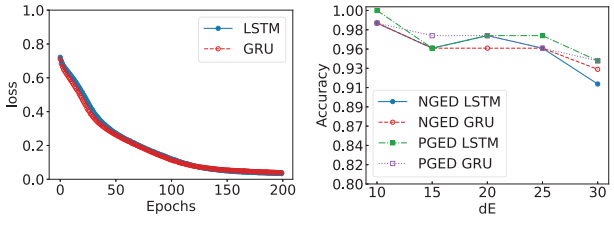
(a) GED computation time (b) GNN computation time

Fig. 6. Computation times for 100 graphs pairwise

We also measured the traditional GED and GCN layers prediction time on a test data set with 100 graphs pairwise using a physical machine with 4-cores Intel's Core 1.6GHz CPU and 16GB RAM. Fig.6 illustrates the obtained results. It is remarkable that GCN layers only need 1 second to compare 100 graphs pairwise with 10 UAVs, which is more than 1000 times faster than the traditional GED method. The reason behind the phenomenon is that GCN layers use neural networks to embed the graph into Euclidean vector space and can compute the similarity directly, which is different from the traditional GED methods that count the number of graph edge

deletions, edge insertion, and vertex relabeling operations to transform one graph to the target one.

3) GNN based spoofing detection approach: We measure two kinds of recurrent networks unit, LSTM, and GRU separately, to evaluate the proposed GNN-based GPS spoofing detection approach. The LSTM and GRU training history is illustrated in Fig.7(a), where the GRU is convergent faster than the LSTM network. The reason for this phenomenon is that the GRU model has a more simple architecture compared to the LSTM. Although LSTM is not far-off from the GRU in detection accuracy, the LSTM requires more training time compared with the GRU. In practice, a longer training time introduces more latency for updating the GNN model in a dynamic environment and may hinder the spoofing attack during model updating. Thus, the combination of GNN and GRU is better than GNN and LSTM for the cellular-connected UAV swarm GPS spoofing detection. In addition to the training performance, we further compare trained LSTM and GRU model performances with the Normalized GED (NGED) and the Predicted GED (PGED) under different GPS error settings as shown in Fig.7(b). From the results depicted in Fig.7(b), the accuracy of GNN is decreasing as GPS error increases. This can be explained by the fact that it is easier for an attacker to counterfeit the GPS position in an environment with a high GPS error. Moreover, we observe that the use of NGED and PGED has insignificant impacts on the GNN spoofing detection performance. The reason behind those trends is that the proposed GNN spoofing detection approach considers not only the swarm spatial similarity but also the temporal changes caused by the spoofing attack.



(a) Training history (b) Detection performance

Fig. 7. Training history and detection performance for 10 UAVs

V. Conclusion and Future Work

This paper proposed a GNN-based GPS spoofing detection approach for the cellular-connected UAV swarm using the similarity between the swarm GPS topology and the communications topology. Specifically, the GCN layers and recurrent network layers are used to compute the spatial and temporal similarity between the swarm GPS location topology and communications topology. Remarkably, GNN can compute two graphs' similarity computation within 10 milliseconds compared with the 16 seconds consumed by the traditional GED approach. Furthermore, the use of the GRU unit requires less training time than the use of the LSTM unit. In the future, we will elaborate more on graphic reinforcement learning-based GPS spoofing detection and mitigation methods for the cellular-connected UAV swarm.

VI. Acknowledgement

This work was supported Academy of Finland project ULTRA under Grant No.328215 and Profi-5 under Grant No.326346.

References

[1] L. Zhou, S. Leng, Q. Wang, and Q. Liu, "Integrated sensing and communication in uav swarms for cooperative multiple targets tracking," *IEEE Transactions on Mobile Computing*, no. 01, pp. 1–17, 2022.

[2] M. Ceccato, F. Formaggio, and S. Tomasin, "Spatial gnss spoofing against drone swarms with multiple antennas and wiener filter," *IEEE Transactions on Signal Processing*, vol. 68, pp. 5782–5794, 2020.

[3] A. Rugo, C. A. Ardagna, and N. E. Ioini, "A security review in the uavnet era: Threats, countermeasures, and gap analysis," *ACM Computing Surveys (CSUR)*, vol. 55, no. 1, pp. 1–35, 2022.

[4] M. Foruhandeh, A. Z. Mohammed, G. Kildow, P. Berges, and R. Gerdes, "Spotr: Gps spoofing detection via device fingerprinting," in *Proceedings of the 13th ACM Conference on Security and Privacy in Wireless and Mobile Networks*, 2020, pp. 242–253.

[5] M. Jayaweera, "A novel deep learning gps anti-spoofing system with doa time-series estimation," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 1–6.

[6] H. Sathaye, G. LaMountain, P. Closas, and A. Ranganathan, "Semperfi: Anti-spoofing gps receiver for uavs," in *Network and Distributed Systems Security (NDSS) Symposium 2022*, 2022.

[7] D.-Y. Jeon, T. Gaybullaev, J. H. Noh, J.-M. Joo, S. J. Lee, and M.-K. Lee, "Performance analysis of authentication protocols of gps, galileo and beidou," *Journal of Positioning, Navigation, and Timing*, vol. 11, no. 1, pp. 1–9, 2022.

[8] Z. Wu, R. Liu, and H. Cao, "Ecdsa-based message authentication scheme for beidou-ii navigation satellite system," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 55, no. 4, pp. 1666–1682, 2018.

[9] Z. Wu, Y. Zhang, and R. Liu, "BD-II NMA&SSI: An Scheme of Anti-Spoofing and Open BeiDou II D2 Navigation Message Authentication," *IEEE Access*, vol. 8, pp. 23 759–23 775, 2020.

[10] M. Nicola, B. Motella, M. Pini, and E. Falletti, "Galileo osnma public observation phase: Signal testing and validation," *IEEE Access*, vol. 10, pp. 27 960–27 969, 2022.

[11] J. Curran and N. Hanley, "On the energy and computational cost of message authentication schemes for gnss," *IEEE Aerospace and Electronic Systems Magazine*, vol. 34, no. 1, pp. 40–53, 2019.

[12] S. Dasgupta, M. Rahman, M. Islam, and M. Chowdhury, "A sensor fusion-based gnss spoofing attack detection framework for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2022.

[13] Y. Dang, C. Benzaïd, Y. Shen, and T. Taleb, "GPS Spoofing Detector with Adaptive Trustable Residence Area for Cellular based-UAVs," in *GLOBECOM 2020-2020 IEEE Global Communications Conference*. IEEE, 2020, pp. 1–6.

[14] Y. Dang, C. Benzaïd, B. Yang, T. Taleb, and Y. Shen, "Deep ensemble learning based gps spoofing detection for cellular-connected uavs," *IEEE Internet of Things Journal*, 2022.

[15] Y. Dang, C. Benzaïd, T. Taleb, B. Yang, and Y. Shen, "Transfer learning based gps spoofing detection for cellular-connected uavs," in *2022 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE, 2022, pp. 629–634.

[16] 3GPP TR 36.777, "Enhanced LTE Support for Aerial Vehicles," Jan. 2018.

[17] D. B. Blumenthal and J. Gamper, "On the exact computation of the graph edit distance," *Pattern Recognition Letters*, vol. 134, pp. 46–57, 2020.

[18] Y. Bai, H. Ding, S. Bian, T. Chen, Y. Sun, and W. Wang, "Simgnn: A neural network approach to fast graph similarity computation," in *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 2019, pp. 384–392.

[19] R. Rana, "Gated recurrent unit (gru) for emotion classification from noisy speech," *arXiv preprint arXiv:1612.07778*, 2016.

[20] R. J. Qureshi, J.-Y. Ramel, and H. Cardot, "Graph based shapes representation and recognition," in *International Workshop on Graph-Based Representations in Pattern Recognition*. Springer, 2007, pp. 49–60.