Attack-Resilient Connectivity Game for UAV Networks using Generative Adversarial Learning

Bo Yang and Min Liu
Institute of Computing Technology, Chinese Academy of Sciences
Beijing, China
yangbo_010@163.com,liumin@ict.ac.cn

ABSTRACT

The continuous link connectivity is critical for the efficient collaboration of multiple unmanned aerial vehicles (UAVs). However, the UAV communication environments are not only harsh, but are also confronted with the threats of smart attackers, which pose great barriers in maintaining the links unblocked. In this paper, we leverage the paradigm of the Generative Adversarial Network (GAN) to formulate an attack-resilient connectivity game between a pair of neighboring UAVs and an attacker. In the three-agent adversary game, the attacker acts as the generator, which attempts to generate highly approximate information as the UAVs so as to maximize its jamming capability; while the pairwise UAVs act as the discriminators, which attempt to enhance the capability of refusing the fake information (i.e., the opponent's attack). As the state-of-the-art GAN learning algorithms suffer from the instability dilemma (i.e., either with the unsuccessful convergence or with the low generation/discrimination performance), we incorporate the conditional GAN with the least square objective loss function as well as the mean square error such that the attacker can improve the detection capability from UAVs' historical activity patterns and the UAVs can accordingly adjust the connectivity strategy. We validate the effectiveness of the proposed algorithm through extensive evaluations. Results demonstrate that the proposed algorithm can improve the convergence efficiency, reduce the connection latency, and enhance the attack-resilience capability significantly.

KEYWORDS

Unmanned aerial vehicles; Connectivity establishment; Smart attacks; Adversarial learning

ACM Reference Format:
Bo Yang and Min Liu. 2019. Attack-Resilient Connectivity Game for UAV Networks using Generative Adversarial Learning. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 9 pages.

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada.

1 INTRODUCTION

Networked unmanned aerial vehicles (UAVs) have emerged as a significant technology for civil, public and military applications such as parcel delivery, environment surveillance, precision agriculture, reconnaissance and peacekeeping. The agile UAVs operating in the free space can provide humans with practical, intelligent and irreplaceable services. However, constrained by the energy, payload, coverage area and execution capability of one single UAV, many of the above applications require multiple drones to fulfill a complicated and time-critical task collaboratively. Thus, the unimpeded information delivery and sharing play a key role in coordinating the multi-UAV networks [29].

However, the continuous connectivity maintenances and the resultant multi-UAV coordinations are confronted with particular difficulties, which are listed as follows:

- Compared with the terrestrial wireless networks, the UAV networks have some distinctive features such as the free flight space, high mobility, harsh and dynamic communication environments. As a result, it is challenging to keep connected with the nearby UAVs over the unstable channels.
- The potential airborne spectrum resources (e.g., IEEE L-band and C-band [21]) are not sufficient for the UAV communications particularly when the UAV swarm coexists with other wireless devices. Thus, the spectrum resource scarcity and the inevitable interferences also reduce the idle channels and hinder the multi-UAV connectivity [14].
- It is common that a variety of UAVs manufactured from different vendors are collected together to fulfill tasks. There always exist obvious gaps in operation mechanisms and sensing capabilities, which are critical obstacles for the multi-UAV coordination.
- The UAV networks are particularly vulnerable to cyber-attacks in the sense that the defense measures in the air are much weaker than the ground. A successful attack on safety-critical UAVs (e.g., the military tactical networks) directly results in the block of communications, information leakage from compromised defectors and even disastrous consequences [5, 9, 10].

In a nutshell, the connected link is a prerequisite for the dynamic networking of autonomous UAVs. Any disconnection will lead to failures of multi-UAV communications [32]. In Figure 1, we illustrate the different connectivity status between a pair of UAVs. Only when the UAVs access a same

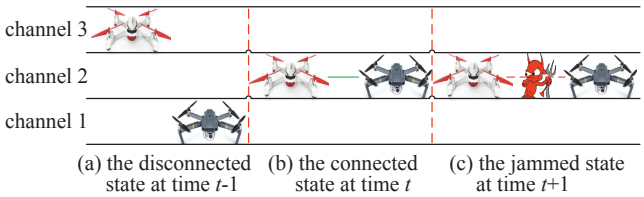

Figure 1: Illustrations of the link status

available and unjammed channel can they get connected to each other.

In this paper, we attempt to design an attack-resilient network connectivity method to facilitate the multi-UAV collaboration. As the opponents aim at maximizing their own attack effects against the legitimate UAVs' defense strategies, we adopt a generative learning model termed Generative Adversarial Networks (GANs) [12]. In the GAN model, there are two separate neural networks with competitive goals: the discriminator aims at distinguishing between the real and the fake information, while the generator aims at generating as realistic information as possible to mislead the discriminator. By iteratively training, the accuracies of both discriminator and generator become more and more sensitive. Analogously, the discriminator can be regarded as the legitimate UAV, which aims at defending against malicious attacks and getting connected to collaborative peers, while the generator can be regarded as the attacker, which aims at disguising itself as the legitimate UAV and blocking the UAVs' normal communications.

However, the standard GAN suffers from the dilemmas of *mode collapse* (i.e., it is difficult to reach the convergence) and *trivial performance* (i.e., the generated fake samples are dissatisfactory). In this work, we introduce a three-agent GAN structure to characterize the adversary between a pair of legitimate UAVs and a malicious jammer. This pair of UAVs (i.e., two discriminators) can jointly identify the interference behavior of the jammer (i.e., the generator) and thus a better anti-jamming capability can be expected. In order to address the *mode collapse* dilemma, we incorporate the conditional GAN with the least square loss function. The conditional GAN is adopted due to the fact that the future accessed channels are relevant to the historical channel switching patterns. Moreover, the least square loss function can achieve a stable training. In order to address the *trivial performance* dilemma, we introduce an empirical objective function that can oblige the generator to produce high-fidelity samples. We evaluate the proposed algorithm on the UAVs' anti-jamming network connectivity task. Results display that our algorithm enables UAVs to reliably keep away from the jammed channels and to quickly reach the channel consensus with the target peers.

2 RELATED WORK

In this section, we review some representative works of adversarial learning in the community of UAV networks.

2.1 Anti-jamming Game for UAVs

As the attackers naturally compete for the communication resources with the legitimate nodes, the game-based anti-jamming methods have become feasible solutions. Chen et al. [4] investigate the connectivity maintenance problem in the Internet of Battlefield Things (IoBTs) under the threats of attackers, and the subgame perfect Nash equilibrium (SPE) is adopted to resist link failures. Abuzainab and Saad [2] also study the connectivity maintenance problem in IoBTs, and the counteractions between the defender and the attacker is casted as the dynamic multi-stage Stackelberg connectivity game. A psychological dynamic game between the soldier and the attacker in IoBTs is investigated in [16]. By estimating each other's belief and behavior, the soldier (*resp.* attacker) can decide the opponent's connection (*resp.* disconnection) strategies. By virtue of the Aoyagi's game theory, [7] presents a group communication system that can maximize the mission effectiveness for resource-restricted military tactical networks. In [1], a non-cooperative combinatorial game is investigated for the interference-resilient channel access problem, and the game is certain to converge to a pure-strategy Nash equilibrium. In [28], a distributed anti-coordination game algorithm is proposed to tackle the channel interference issue in the coexisting networks of UAVs and D2D devices. In [31], a prospect-theory-based static game is studied between the subjective attacker and the UAV transmission. Moreover, the reinforcement-learning-based dynamic game is also studied by interacting with the unknown environments repeatedly.

2.2 GAN-based Applications in Networks

The GAN model has been widely applied in image and text generation tasks. However, the investigations of GANs on communication networks are still in infancy. In [30], the AdvGAN framework is proposed to produce adversarial perturbations efficiently. Thus, AdvGAN performs well in improving adversarial defense methods. In [23], the conflicting privacy objectives between the defense mechanism and the inference attack are modeled by the GAN. In [8], the adversarial learning is leveraged to generate high-fidelity synthetic training data so as to improve the accuracy of spectrum sensing in the dynamic environment. In [6], the GAN model is utilized for data-driven scenario generation of renewable resources, which is significant for decision-making in power grids. In [18], Lei et al. apply the GAN to generate high-quality graph topologies and to tackle the temporal link prediction of dynamic networks. In [34], the GAN model is utilized to infer city-wide fine-grained mobile data traffic consumptions from limited coarse-grained measurements collected by probes.

In this paper, we utilize the GAN technique to characterize the adversary between the legitimate UAVs and the malicious jammer, trying to maintain the UAVs' connected links under the threats of smart attacks.

3 PRELIMINARIES ON GAN

The GAN [12] learns a generative model as a two-player zero-sum game between a generator and a discriminator:

- Generator G: G tries to generate (or select) the target's information from the candidate resource pool for the given simulation task. Its objective is to approximate the real information distribution as much as possible.
- Discriminator D: D tries to distinguish the real objects (or samples) from the fake ones. Actually, it is a binary classifier, which could be labelled as 1 if the objects (or samples) are real and could be labelled as 0 otherwise.

The adversarial competition between G and D can be denoted as the following minimax objective function:

$$\min_{\theta_G}\max_{\theta_D}\mathcal{V}(G,D)=\min_{\theta_G}\max_{\theta_D}\mathbb{E}_{x\sim p(x)}\log D\left(x;\theta_D\right)+\mathbb{E}_{z\sim p(z)}\log\left(1-D\left(G\left(z;\theta_G\right);\theta_D\right)\right), \quad (1)$$

where θ_G and θ_D denote the learning parameters of the generator G and the discriminator D respectively, G takes a noise z from a noise distribution $p(z)$ as the input and outputs a sample, and $p(x)$ denotes the real distribution of the training data x. For brevity, we use $G(z)$ (*resp.* $D(x)$) instead of $G(z;\theta_G)$ (*resp.* $D\left(x;\theta_D\right)$) in the rest of the paper.

The objective of D is to minimize the loss function of $\mathcal{L}(D)=-(\mathbb{E}_{x\sim p(x)}\log D(x)+\mathbb{E}_{z\sim p(z)}\log(1-D(G(z))))$ while fixing G. Analogously, the objective of G is to minimize the loss function of $\mathcal{L}(G)=\frac{1}{|D_z|}\sum_i[\log(1-D(G(z^{(i)})))]$ while fixing D.

Initially, G has poor ability in approximating the real samples, D can thus resist the fake samples confidently. As the training proceeds, both the counterfeiting capability of G and the identification capability of D enhance, and this property can be utilized for adversarial learning tasks.

The GAN paradigm is suitable for tackling the competition tactics. However, the standard GAN training provides no convergence guarantee when finding a Nash equilibrium of a non-cooperative game with high-dimensional and continuous parameters [26].

To this end, the ongoing efforts aim at achieving the stable and efficient training, particularly for optimizing the loss functions. For examples, Mao et al. [20] utilize the least square loss functions (i.e., $\mathcal{L}(D)_{LSGAN}=-\mathbb{E}_{x\sim p(x)}[(D\left(x\right)-1)^2]+\mathbb{E}_{z\sim p(z)}[D(z)^2]$ and $\mathcal{L}(G)_{LSGAN}=-\mathbb{E}_{z\sim p(z)}[D(z-1)^2]$) for stable training. Thus, the training objective is to minimize the Pearson χ^2. The Wasserstein GAN (WGAN) [3] defines the close distance between the real distribution and the generated distribution (i.e., $\rho(p(x),p(z))$). The loss functions of WGAN are $\mathcal{L}(D)_{WGAN}=-\mathbb{E}_{x\sim p(x)}D\left(x\right)+\mathbb{E}_{z\sim p(z)}D\left(G\left(z\right)\right)$ and $\mathcal{L}(G)_{WGAN}=-\mathbb{E}_{z\sim p(z)}D\left(G\left(z\right)\right)$ respectively. However, the weight clipping method in WGAN may lead to low-quality generation samples and the non-convergence dilemma. As a countermeasure, the WGAN-GP model [13] adds a soft constraint with a penalty on the gradient norm. Compared with the $\mathcal{L}(D)_{WGAN}$, the $\mathcal{L}(D)$ of WGAN-GP adds an item of $\lambda\mathbb{E}_{z\sim p(z)}[(\parallel\nabla_z D(z)\parallel_2-1)^2]$. Hence, WGAN-GP yields a more stable training. The CycleGAN model [35] adds a cycle consistency loss that enables symmetric bijections of $F(G(x))\approx x$ and $G(F(z))\approx z$, and the corresponding inverse loss function is $\mathcal{L}_{Cyc}=\mathbb{E}_{x\sim p(x)}[\parallel F(G(x))-x\parallel_1]+\mathbb{E}_{z\sim p(z)}[\parallel G(F(z))-z\parallel_1]$. CycleGAN is suitable for the case with no paired training data. Combined with WGAN [3] and CycleGAN [35], the CWR-GAN model [22] adds a regression loss on the paired samples, and thus can prohibit the mode-collapse. Coincidentally, the VEE-GAN model [27] avoids the mode-collapse by incentivizing the newly-built reconstructor network to map all real data to the noise distribution, the inverse approximation of the generator network can thus produce high-quality samples.

Another roadmap of optimizing the adversarial learning is to redesign the architectures of GANs. For examples, the Triple-GAN model [19] conducts the three-player game by adding a classifier, the conditional training of the generator and the classifier enables them to reach their own optima. The D2GAN model [25] aims at enhancing the identification capability, and thus the game is conducted among two discriminators and one generator by minimizing two divergences. Moreover, D2GAN can reach the global optimum provided that $p(z)=p(x)$. The MAD-GAN model [11] incorporates multiple generators to generate diverse and high-quality samples to mislead the only discriminator. MAD-GAN also modifies the discriminator's objective function so as to identify the exact generator from confusing samples.

The algorithm in this paper is a combination of the above two roadmaps: we optimize the loss function for an efficient adversarial training; meanwhile, we adopt an appropriate architecture to characterize the conflict between the pairwise communication agents of sender/receiver (i.e., two discriminators) and the malicious jammer (i.e., the generator).

4 GAN-BASED ATTACK-RESILIENT CONNECTIVITY GAME

The anti-jamming link connectivity task concerns how to access the appropriate channel to set up a connection to another neighboring UAV in the jamming-threatening tactical network. In this section, we present the GAN-based algorithm for the anti-jamming connectivity problem.

4.1 System Model

We consider a distributed flying ad-hoc network (FANET) consisting of multiple collaborative UAVs. The set of UAVs is denoted as $\mathcal{U}$. Each UAV has the capability of perceiving the channel status and switching across different channels. The UAVs are operating on a set of N orthogonal channels. Due to the different locations and perception capabilities, the sensed available channels at each UAV can be arbitrary. Let C_s and C_r denote the available channel set of the sender node s and the receiver node r respectively ($\forall s,r\in\mathcal{U}$).

As the data transmission among the collaborative UAVs relies on the prerequisite of wireless connectivity, the sender

s must firstly select and access the same qualified idle channel from multiple candidate channels as the target receiver r and set up a link. Thus, node s (*resp.* r) has to send (*resp.* receive) tentative announcement messages and fulfils the handshaking process. Let $c_s^t \in C_s$ and $c_r^t \in C_r$ denote the channels that nodes s and r access at the time instant t respectively. Over a continuous period of time, the channel switching pattern of each node forms a sequence of channels. Take node s as example, its channel switching pattern is denoted as $S_s = \{c_s^1, c_s^2, \ldots, c_s^t, \ldots\}$. Moreover, to ensure the connected link, we assume that the available channels between any pair of UAVs can be different but overlap (i.e., $\forall s, r \in \mathcal{U}, C_s \cap C_r \neq \emptyset$). Only when $c_s^t = c_r^t$ and this common channel is qualified can this pair of UAVs transmit data packets.

Besides the co-located UAVs, we assume that a limited number of malicious jammers (either attackers or compromised nodes) also exist in the FANET. The jammers are uncoordinated and share no information among them. Some jammers misbehave directly under the aim of interfering the legitimate communications and draining the energies of UAVs, while some jammers behave normally but eavesdrop. In this paper, we focus on the former scenario, which further includes two types: *Reactive jamming* and *Fake ACK attack* [15]. Specifically, the sender and the receiver need to agree on the common available channel by exchanging the tentative announcement messages. The reactive jammer jams a channel only when it detects an on-going message transmission so as to save energy. If the jammer intercepts the messages successfully, it will transmit the generated fake acknowledgement message to the sender. Consequently, the sender is deceived by receiving a fake acknowledgement message from the jammer. But in fact, the receiver does not receive the real message at all. As a result, the communication latency is prolonged and the link is disrupted by the jammer.

This connectivity game is suitable to be tackled by GAN due to the fact that the sender and the receiver have the common objective of channel consensus against the jammer's interference behavior. In the GAN framework, the sender and the receiver can be modeled as the discriminators, which aim to reach the channel consensus as quickly as possible while keeping away from the jammed channel. While the jammer corresponds to the generator, which tries to mislead and block the legitimate nodes by transmitting fake messages.

4.2 Three-agent Adversarial Connectivity Learning Algorithm

We formulate a three-agent game that includes *Double discriminators* of D_s (i.e., sender) and D_r (i.e., receiver), and one *Single generator* of G (i.e., jammer), we thus term the presented algorithm as DS-GAN.

Considering the common objective of an agreed idle channel between the sender and the receiver, the holistic objective function of DS-GAN can be decomposed as:

$$\begin{aligned}\min_G \max_{D_s, D_r} \mathcal{V}(G, D_s, D_r) =& \lambda_s \min_G \max_{D_s} \mathcal{V}(G, D_s) + \\ & \lambda_r \min_G \max_{D_r} \mathcal{V}(G, D_r),\end{aligned} \tag{2}$$

where λ_s and λ_r denote the factor weights of D_s and D_r respectively.

4.2.1 Designing the Loss Functions. To get connected to the target neighbor, nodes need to transmit announcement messages over different available channels tentatively, which activate the detection behavior of the jammer. The jammer attempts to take control of the nodes' channel switching patterns and to forge the legitimate nodes' identities.

Initially, the jammer has no knowledge of the nodes' switching patterns, the legitimate nodes can avoid to access the jammed channel with high confidence. However, the jammer gradually gets aware of the nodes' switching patterns by detecting and deriving the past accessed channels. That is, the potential jamming attack is conditioned on the historical observations. Thus, we adopt the conditional GAN [24] for the security game.

Here, we build a connection between the channel labels (i.e., the given classes) and the generated samples. The objective of G is to predict and generate the samples under the observed channel labels. The adversary between the sender and the jammer is formulated as:

$$\begin{aligned}\mathcal{L}_{cGAN}(G, D_s) =& \min_G \max_{D_s} \mathbb{E}_{x_s \sim p(x_s)} \log D_s\left(x_s|y\right) + \\ & \mathbb{E}_{z \sim p(z)} \log\left(1 - D_s\left(G\left(z|y\right)\right)\right),\end{aligned} \tag{3}$$

where x_s corresponds to the sender's potential channel switching pattern $S_s = \{c_s^t, \ldots\}$ ($c_s^t \in C_s, s \in \mathcal{U}$), y corresponds to the previous channel switching pattern $S_s' = \{\ldots, c_s^{t-1}\}$[1], and z corresponds to the jammer's potential channel switching pattern.

However, the standard conditional GAN suffers from the dilemma of instability. As the LSGAN model [20] provides the discriminator with a smooth and non-saturating gradient, and aims at making the real and generated samples indistinguishable, we thus incorporate LSGAN with the conditional GAN for a stable training. The suggested loss function combines the least square loss with that of the conditional GAN. The loss function of D_s is given as:

$$\begin{aligned}\mathcal{L}_{cLSGAN}(D_s) =& \mathbb{E}_{x_s \sim p(x_s)}[\left(D_s\left(x_s|y\right) - 1\right)^2] + \\ & \mathbb{E}_{z \sim p(z)}[\left(D_s\left(G\left(z|y\right)\right) + 1\right)^2].\end{aligned} \tag{4}$$

Likewise, the conditional loss function of D_r is given as:

$$\begin{aligned}\mathcal{L}_{cLSGAN}(D_r) =& \mathbb{E}_{x_r \sim p(x_r)}[\left(D_r\left(x_r|y\right) - 1\right)^2] + \\ & \mathbb{E}_{z \sim p(z)}[\left(D_r\left(G\left(z|y\right)\right) + 1\right)^2],\end{aligned} \tag{5}$$

where x_r corresponds to the receiver's potential channel switching pattern $S_r = \{c_r^t, \ldots\}$ ($c_r^t \in C_r, r \in \mathcal{U}$).

[1] As the sender/receiver's historical channel switching patterns can both be detected by the jammer, we thus omit y's subscript of s/r.

Correspondingly, the conditional loss function of G from the perspective of D_s is given as:

$$\mathcal{L}_{cLSGAN}(G) = \mathbb{E}_{z\sim p(z)}[(D_s\left(G(z)|y\right))^2]. \tag{6}$$

To enhance G's capabilities of detection and forgery, Eqn. (6) is extended by an additional loss term:

$$\mathcal{L}_{L_1}(G) = \mathbb{E}_{x_s\sim p(x_s)}[\| \, y - G(x_s, z) \, \|_1], \tag{7}$$

which obliges the generator to produce high-fidelity samples.

In order to address the dilemma of trivial performance, the generated samples should be encouraged to be close to the real data. A straightforward way to achieve this objective is to minimize the Euclidean distance between the generated and the real data distributions, resulting in the additional term of mean square error (MSE) to G's loss function:

$$\mathcal{L}_{MSE}(x_s, G(z;\theta)) = \| \, x_s - G(z;\theta) \, \|_2^2 \, . \tag{8}$$

Specifically, the optimal $\hat{\theta}$ can be obtained by solving the following formula:

$$\begin{aligned} \hat{\theta} &= \arg\min_{\theta} \frac{1}{T}\mathcal{L}_{MSE}(x_s, G(z;\theta)) \\ &= \arg\min_{\theta} \frac{1}{T}\sum_{t}^{T} \left\| x_s^t - G\left(z^t;\theta\right)\right\|_2^2. \end{aligned} \tag{9}$$

Eqn. (9) can oblige the fake samples to be close to the real ones while the adversarial loss aims to minimize the divergence between these two distributions.

Putting Eqns. (6)-(9) all together, the final objective loss of G from the perspective of D_s is

$$\begin{aligned} \mathcal{L}_{D_s}(G) =& \mathcal{L}_{cLSGAN}(G, D_s) + \lambda_1\mathcal{L}_{L_1}(G) + \lambda_2\mathcal{L}_{MSE} \\ =& \mathbb{E}_{z\sim p(z)}[(D_s\left(G(z)|y\right))^2]+ \\ & \lambda_1\mathbb{E}_{x_s\sim p(x_s)}[\| \, y - G(x_s, z) \, \|_1]+ \\ & \lambda_2\frac{1}{T}\sum_{t}^{T} \left\| x_s^t - G\left(z^t\right)\right\|_2^2, \end{aligned} \tag{10}$$

where λ_1 and λ_2 denote the balancing parameters.

As for the adversary between G and D_r, it can be obtained in the same way as Eqn. (10), which is given as:

$$\begin{aligned} \mathcal{L}_{D_r}(G) =& \mathcal{L}_{cLSGAN}(G, D_r) + \lambda_3\mathcal{L}_{L_1}(G) + \lambda_4\mathcal{L}_{MSE} \\ =& \mathbb{E}_{z\sim p(z)}[(D_r\left(G(z)|y\right))^2]+ \\ & \lambda_3\mathbb{E}_{x_r\sim p(x_r)}[\| \, y - G(x_r, z) \, \|_1]+ \\ & \lambda_4\frac{1}{T}\sum_{t}^{T} \left\| x_r^t - G\left(z^t\right)\right\|_2^2, \end{aligned} \tag{11}$$

where λ_3 and λ_4 denote the balancing parameters.

In view of Eqns. (2), (10) and (11), it is apparent to obtain the holistic objective loss function of G against both D_s and D_r.

4.2.2 Training the DS-GAN. The training process is formally described in Algorithm 1, which utilizes the Stochastic Gradient Descent (SGD) idea to update the DS-GAN parameters.

Specifically, we adopt Adam stochastic approximation [17] to avoid the undesired case of *mode collapse* (i.e., the generated samples are lack of diversity and deviate from the real distribution). Adam leverages the parameters of $(\beta_1;\beta_2,\alpha)$ to average the (squared) gradient, which indicates an impulse to push the generator towards various domains.

Algorithm 1: The DS-GAN training algorithm

Input: the batch size m; the iteration numbers for G, D_s and D_r (i.e., n_G and n_D); the factor weights of λ_s and λ_r; the balancing parameters of $\lambda_1, \cdots, \lambda_4$; the Adam hyper-parameters of α, β_1, β_2; the learning rate η.

Initialize two discriminators D_s, D_r and the generator G parameterized by $\theta_{D_s}, \theta_{D_r}$ and θ_G respectively;

while $\theta_{D_s}, \theta_{D_r}$ *and* θ_G *are not convergent* **do**

  for $epoch_D = 1$ *to* n_D **do**

    Sample batch from channel switching patterns $\{x_s^t, y^t\}_{t=1}^m, x_s^t, y^t \subseteq S_s, s \in \mathcal{U}$;

    Sample batch from channel switching patterns $\{x_r^t, y^t\}_{t=1}^m, x_r^t, y^t \subseteq S_r, r \in \mathcal{U}$;

    Update D_s by using gradient descent:

$$g_{D_s} \leftarrow \nabla_{D_s}[\frac{1}{m}\sum_{t=1}^{m}\left(D\left(x_s^t\left|y^t\right.\right)-1\right)^2+ \frac{1}{m}\sum_{t=1}^{m}(D\big(G\left(z^t\left|y^t\right.\right)\big)+1)^2];$$

    $\theta_{D_s} \leftarrow \theta_{D_s} + \eta \cdot Adam(\theta_{D_s}, g_{D_s}, \alpha, \beta_1, \beta_2)$;

    Update D_r by using gradient descent:

$$g_{D_r} \leftarrow \nabla_{D_r}[\frac{1}{m}\sum_{t=1}^{m}\left(D\left(x_r^t\left|y^t\right.\right)-1\right)^2+ \frac{1}{m}\sum_{t=1}^{m}(D\big(G\left(z^t\left|y^t\right.\right)\big)+1)^2];$$

    $\theta_{D_r} \leftarrow \theta_{D_r} + \eta \cdot Adam(\theta_{D_r}, g_{D_r}, \alpha, \beta_1, \beta_2)$;

  for $epoch_G = 1$ *to* n_G **do**

    Sample batch from channel switching patterns $\{z^t, y^t\}_{t=1}^m$;

    Update G by using gradient descent:

$$g_G \leftarrow \nabla_G[\frac{1}{m}\sum_{t=1}^{m}\left(D\left(G(z^t)\left|y^t\right.\right)\right)^2+ \lambda_{1\ (\text{or } 3)}\frac{1}{m}\sum_{t=1}^{m}\left\|y^t - G\left(x_{s\ (\text{or } r)}^t, z^t\right)\right\|_1+ \lambda_{2\ (\text{or } 4)}\frac{1}{m}\sum_{t=1}^{m}\| \, x_{s\ (\text{or } r)}^t - G(z^t) \, \|_2^2];$$

    $\theta_G \leftarrow \theta_G - \eta \cdot Adam(\theta_G, g_G, \alpha, \beta_1, \beta_2)$;

In the three-agent adversary game, the pairwise discriminators and the generator are trained iteratively for n_D and n_G times respectively. Unless the objective loss functions are convergent, the discriminators' parameters will be adjusted by fixing the opponent's parameters, and vice versa. In each iteration, D_s, D_r and G stochastically sample m potential and previous channels, and calculate their respective gradients g_{D_s}, g_{D_r} and g_G according to the corresponding loss functions. These gradients are then utilized to adjust the neural network parameters $\theta_{D_s}, \theta_{D_r}$ and θ_G.

By performing the iterative training algorithm, the anti-jamming intercommunication capability of the UAVs and the jamming capability of the attacker are improved synchronously. Particularly, armed with the empirical loss function of MSE, the jammer can track the UAVs' channel switching patterns with minor deviations. Meanwhile, the UAVs also learn to identify the attack behaviors with high accuracy and take adaptive measures to keep away from the jammed channels by virtue of the least square loss.

4.2.3 Theoretical Analysis. To analyze the equilibrium condition of the minimax game in the proposed DS-GAN model, we first consider the optimal discriminators (i.e., $D_s^\star$ and $D_r^\star$) under an arbitrary generator G.

Proposition 4.1. *For any fixed generator G, maximizing $\mathcal{V}(G, D_s, D_r)$ yields to the following closed-form optimal discriminators $D_s^\star$ and $D_r^\star$:*

$$D_s^\star = \frac{p(x_s)-p(x_{zs})}{p(x_s)+p(x_{zs})} \text{ and } D_r^\star = \frac{p(x_r)-p(x_{zr})}{p(x_r)+p(x_{zr})}.$$

Proof. Given a fixed G, the learning criteria for D_s is to minimize

$$\begin{aligned}\mathcal{V}(G, D_s) =& \mathbb{E}_{x_s\sim p(x_s)}[(D_s(x_s|y)-1)^2]+ \\ & \mathbb{E}_{z\sim p(z)}[(D_s(G(z|y))+1)^2] \\ =& \int_\chi (p(x_s)(D_s(x_s|y)-1)^2+ \\ & p(x_{zs})(D_{zs}(x_{zs}|y)+1)^2)dx.\end{aligned}$$

While fixing G, the loss function of D_s is minimized if the generated samples approximate to the real data. In this case, the jammer's potential channel switching sequence z and the sender's potential channel switching sequence x_s follow similar patterns. Thus, $D_s(G(z|y))$ can be regarded as $D_{zs}(x_{zs}|y)$.

Considering the internal function of $(p(x_s)(D_s(x_s|y)-1)^2+p(x_{zs})(D_{zs}(x_{zs}|y)+1)^2)$, it reaches the minimum value at $\frac{p(x_s)-p(x_{zs})}{p(x_s)+p(x_{zs})}$ with respect to D_s.

Similarly, the training function of D_r reaches its minimum at $\frac{p(x_r)-p(x_{zr})}{p(x_r)+p(x_{zr})}$. This concludes the proof. □

Theorem 4.2. *Optimizing the DS-GAN model is equivalent to minimize the Pearson χ^2 divergence between $p(x_s)+p(x_{zs})$ and $2p(x_{zs})$ with respect to D_s (resp. $p(x_r)+p(x_{zr})$ and $2p(x_{zr})$ with respect to D_r).*

Proof. Substituting the optimal $D_s^\star$ and $D_r^\star$ in Proposition 1 into Eqn. (2), the minimax game can be reformulated as:

$$\begin{aligned}G^\star =& \arg\min_G \mathcal{V}(G, D_s^\star, D_r^\star) \\ =& \lambda_s(2\cdot \mathrm{JSD}(p(x_s) \parallel p(x_{zs})) - \log 4)+ \\ & \lambda_r(2\cdot \mathrm{JSD}(p(x_r) \parallel p(x_{zr})) - \log 4),\end{aligned} \tag{12}$$

where JSD ($0 \leqslant \mathrm{JSD} \leqslant 1$) denotes the Jensen-Shannon divergence between the generated and the real data distributions.

According to Proposition 4.1, Eqn. (12) can be reformulated as

$$\begin{aligned}G^\star =& \lambda_s \cdot 2\mathbb{E}_{x_s\sim p(x_s)}\left[(D_s^\star(x_s|y))^2\right] + \\ & \lambda_r \cdot 2\mathbb{E}_{x_r\sim p(x_r)}\left[(D_r^\star(x_r|y))^2\right] \\ =& \lambda_s \cdot 2\mathbb{E}_{x_s\sim p(x_s)}\left[\left(\frac{p(x_s)-p(x_{zs})}{p(x_s)+p(x_{zs})}\right)^2\right] + \\ & \lambda_r \cdot 2\mathbb{E}_{x_r\sim p(x_r)}\left[\left(\frac{p(x_r)-p(x_{zr})}{p(x_r)+p(x_{zr})}\right)^2\right] \\ =& \lambda_s \cdot 2\int_\chi p(x_s)\left(\frac{p(x_s)-p(x_{zs})}{p(x_s)+p(x_{zs})}\right)^2 dx+ \\ & \lambda_r \cdot 2\int_\chi p(x_r)\left(\frac{p(x_r)-p(x_{zr})}{p(x_r)+p(x_{zr})}\right)^2 dx \\ =& \lambda_s \cdot 2\int_\chi \frac{((p(x_s)+p(x_{zs}))-2p(x_{zs}))^2}{p(x_s)+p(x_{zs})}dx+ \\ & \lambda_r \cdot 2\int_\chi \frac{((p(x_r)+p(x_{zr}))-2p(x_{zr}))^2}{p(x_r)+p(x_{zr})}dx \\ =& 2\lambda_s\chi^2_{\mathrm{Pearson}}(p(x_s)+p(x_{zs}) \parallel 2p(x_{zs}))+ \\ & 2\lambda_r\chi^2_{\mathrm{Pearson}}(p(x_r)+p(x_{zr}) \parallel 2p(x_{zr})),\end{aligned} \tag{13}$$

where $\chi^2_{\mathrm{Pearson}}$ denotes the Pearson χ^2 divergence.

As a conclusion, optimizing the DS-GAN model is equivalent to minimize the Pearson χ^2 divergence between $p(x_s)+p(x_{zs})$ and $2p(x_{zs})$ with respect to D_s (*resp.* $p(x_r)+p(x_{zr})$ and $2p(x_{zr})$ with respect to D_r). Furthermore, the global minimum of Eqn. (13) is yielded if $p(x_s) = p(x_{zs})$ and $p(x_r) = p(x_{zr})$. □

Remark 4.3. When $p(x_s)+p(x_{zs}) = 0, p(x_s) > p(x_{zs}), p(x_r)+p(x_{zr}) = 0$ and $p(x_r) > p(x_{zr})$, the objective function in Eqn. (13) will become infinite. However, the infinity does not exist in the χ^2 divergence due to the fact that all of $p(x_s), p(x_{zs}), p(x_r)$ and $p(x_{zr})$ are non-negative. As a result, the least square loss function in the proposed DS-GAN model can alleviate the mode collapse dilemma.

5 EXPERIMENTS

In this section, we conduct experiments to evaluate the DS-GAN algorithm.

We select two GAN-based algorithms with three players (i.e., Triple-GAN [19] and D2GAN [25]) and one non-GAN-based game algorithm (i.e., CORE [1]) as the baselines. We apply these algorithms to the UAVs' anti-jamming link connectivity task and utilize three evaluation metrics, which are defined as follows:

- **Average connection latency** – The average time elapsed before the UAVs successfully access a common unjammed channel and set up a link.
- **Attack probability** – The probability that the attacker (i.e., the generator) accurately track the UAVs' activity patterns and impose jammings. This metric can also be interpreted as the ratio between the number of the possibly jammed channels and the number of UAVs' mutual available channels.
- **Packet delivery ratio** – The ratio between the UAVs' successfully delivered data packets and their actually delivered data packets.

We consider a circular flight area with the radius of 100m, in which the UAVs are flying at the same altitude of 50m and at a constant speed of 9m/s. We utilize the free-space outdoor model [33] to characterize the channel status. The UAVs are trying to discover the neighboring node by switching across different channels and broadcasting the announcement messages. While the attacker aims at blocking the UAVs' communication link. Each UAV can only access one channel at a time; likewise, the attacker can only jam one channel at a time.

We divide the experiments into two sets: (1) fixing the total number of channels (i.e., N) and varying the number of UAVs' mutual available channels (i.e., $M = |C_s \cap C_r|$ $(\forall s, r \in \mathcal{U})$); (2) fixing M and varying N.

In the first set of experiments, we assume $N = 50, |C_s| = |C_r| = 0.5N, C_s \neq C_r$ and $M \in [5, 14]$. Figures 2∼4 display the evaluation results under these parameters. Each result is obtained by averaging over 1,000 independent runs of the compared algorithms.

Figure 2 shows the comparisons on the average connection latency. It can be seen that all algorithms cost shorter connectivity time as the value of M increases due to the fact that a larger value of M indicates more opportunities for the channel consensus. The CORE algorithm needs longer connectivity latencies than the other three GAN-based algorithms under all values of M, which indicates that the GAN paradigm is good at the adversary game. Among the three GAN-based algorithms, the D2GAN algorithm performs the worst due to the lack of optimizing the objective loss function. The Triple-GAN algorithm performs better than D2GAN due to the fact that the cross-entropy loss adopted in Triple-GAN can stabilize its convergence and improve the training efficiency. The proposed DS-GAN algorithm yields the shortest average connection latencies under all values of M in the sense that the conditional GAN with the least square objective loss function as well as the mean square error can yield fast convergence.

Figure 3 shows the comparisons on the attack probability. It can be seen that the attack probability will reduce as the value of M increases. Generally, the anti-jamming capability of the CORE algorithm is the weakest. The other GAN-based algorithms can enhance the UAVs' defence capabilities obviously. However, for the D2GAN algorithm, its anti-jamming performance is dissatisfactory, which implies

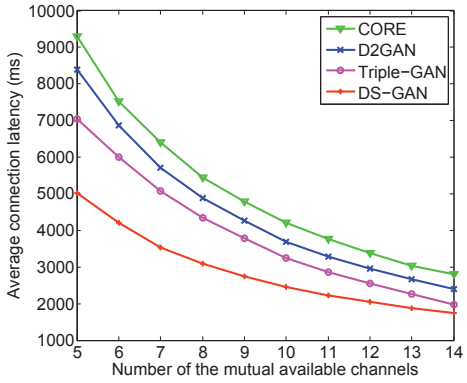

Figure 2: Comparisons on the average connection latency when fixing N and varying M

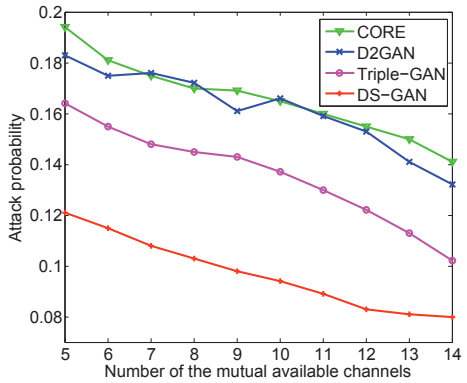

Figure 3: Comparisons on the attack probability when fixing N and varying M

that the accuracy of D2GAN's discriminator is not high. Although the Triple-GAN algorithm's anti-jamming capability is better than D2GAN, Triple-GAN is still dominated by the proposed DS-GAN algorithm. The reason is that DS-GAN's optimized loss function effectively enables the discriminators to refuse fake information.

Figure 4 shows the comparisons on the packet delivery ratio. The variation trends of the packet delivery ratio are relatively stable under different values of M. A larger value of M (i.e., more opportunities for connectivity) results in a slight increase in packet delivery ratio. The DS-GAN algorithm yields the highest packet delivery ratio, followed by the Triple-GAN, D2GAN and CORE algorithms in a descending order.

In the second set of experiments, we assume $N \in [10, 50]$, $|C_s| = |C_r| = 0.5N, C_s \neq C_r$ and $M = 5$. Figures 5∼7 display the evaluation results under these parameters.

Specifically, Figure 5 displays the comparisons on the average connection latency. In contrast to Figure 2, all algorithms cost longer time to get connected as the value of N increases due to the fact that a larger value of N increases the difficulty for the channel consensus. The DS-GAN algorithm yields

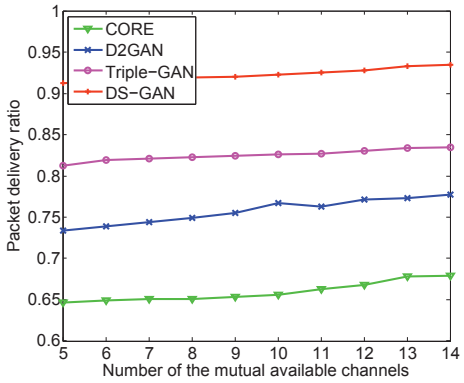

Figure 4: Comparisons on the packet delivery ratio when fixing N and varying M

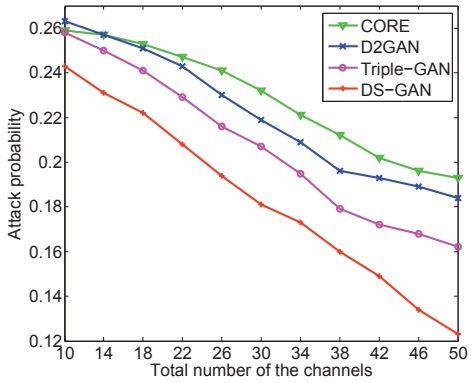

Figure 6: Comparisons on the attack probability when fixing M and varying N

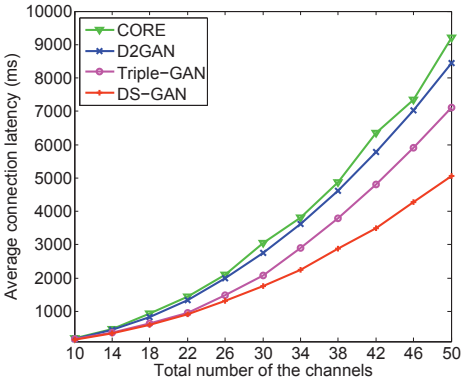

Figure 5: Comparisons on the average connection latency when fixing M and varying N

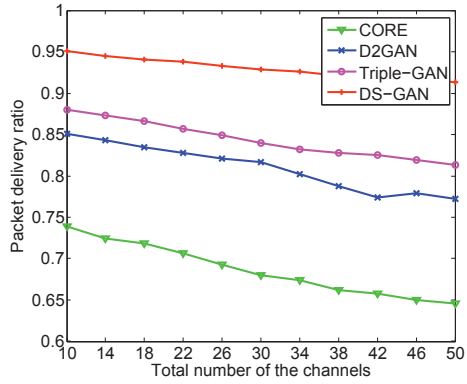

Figure 7: Comparisons on the packet delivery ratio when fixing M and varying N

the shortest average connectivity latencies under all values of N, followed by the Triple-GAN, D2GAN and CORE algorithms in an ascending order.

Figure 6 displays the comparisons on the attack probability. It can be seen that a larger value of N can reduce the attacker's detection accuracy obviously and the DS-GAN algorithm yields the best anti-jamming capability.

Figure 7 displays the comparisons on the packet delivery ratio. The ratio of $\frac{M}{N}$ will decrease when N gets larger. Accordingly, the less connectivity opportunities lead to a smaller packet delivery ratio. However, the downward trends are relatively slow. Above all, the DS-GAN algorithm exhibits apparent advantages on the packet delivery ratio under all values of N.

6 CONCLUSIONS

In this paper, we investigate the UAVs' secure link connectivity problem in the jamming-threatening FANET. We leverage the GAN paradigm to characterize the adversary between a pair of UAVs and a malicious jammer. The usage of the three-agent GAN here is intractable due to the fact that the instable training cannot support the timely message delivery among UAVs and the trivial training performance cannot depict the severe tactical environments. Considering that the potential jamming behavior is conducted by detecting the UAVs' historical channel switching patterns, we incorporate the conditional GAN with the least square objective loss function as well as the mean square error to improve the training performance. Evaluation results also reveal that the UAVs can reliably get connected over the idle channel even confronting with the potential jamming threats.

ACKNOWLEDGMENTS

This work is supported in part by NSFC under Grant 61732017 and in part by CPSF under Grant 2018M631582. Min Liu is the contact author.

REFERENCES

[1] Mohammad J. Abdel-Rahman and Marwan Krunz. 2015. CORE: A combinatorial game-theoretic framework for coexistence rendezvous in DSA networks. In *Proceedings of the 12th IEEE International Conference on Sensing, Communication, and Networking*. IEEE, 10–18.

[2] Nof Abuzainab and Walid Saad. 2018. Dynamic connectivity game for adversarial internet of battlefield things systems. *IEEE Internet of Things Journal* 5, 1 (2018), 378–390.

[3] Martin Arjovsky, Soumith Chintala, and Leon Bottou. 2017. Wasserstein generative adversarial networks. In *Proceedings of 34th International Conference on Machine Learning*. 214–223.

[4] Juntao Chen, Corinne Touati, and Quanyan Zhu. 2017. Heterogeneous multi-layer adversarial network design for the IoT-enabled infrastructures. In *Proceedings of IEEE Global Communications Conference*. IEEE, 1–6.

[5] Xin Chen, Jin-Hee Cho, and Sencun Zhu. 2014. GlobalTrust: An attack-resilient reputation system for tactical networks. In *Proceedings of the 11th IEEE International Conference on Sensing, Communication, and Networking*. IEEE, 275–283.

[6] Yize Chen, Yishen Wang, Daniel Kirschen, and Baosen Zhang. 2018. Model-free renewable scenario generation using generative adversarial networks. *IEEE Transactions on Power Systems* 33, 3 (2018), 3265–3275.

[7] Jin-Hee Cho. 2015. Tradeoffs between trust and survivability for mission effectiveness in tactical networks. *IEEE Transactions on Cybernetics* 45, 4 (2015), 754–766.

[8] Kemal Davaslioglu and Yalin E. Sagduyu. 2018. Generative adversarial learning for spectrum sensing. In *Proceedings of IEEE International Conference on Communications*. IEEE, 1–6.

[9] Fan Fei, Zhan Tu, Ruikun Yu, Taegyu Kim, Xiangyu Zhang, Dongyan Xu, and Xinyan Deng. 2018. Cross-layer retrofitting of UAVs against cyber-physical attacks. In *Proceedings of IEEE International Conference on Robotics and Automation*. IEEE, 550–557.

[10] Amin Ghafouri, Yevgeniy Vorobeychik, and Xenofon Koutsoukos. 2018. Adversarial regression for detecting attacks in cyber-physical systems. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. 3769–3775.

[11] Arnab Ghosh, Viveka Kulharia, Vinay Namboodiri, Philip H.S. Torr, and Puneet K. Dokania. 2018. Multi-agent diverse generative adversarial networks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 8513–8521.

[12] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio *et al.*. 2014. Generative adversarial nets. In *Advances in Neural Information Processing Systems*. 2672–2680.

[13] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. 2017. Improved training of wasserstein GANs. In *Advances in Neural Information Processing Systems*. 1–11.

[14] Lars Hanschke, Leo Kruger, Thomas Meyerhoff, Christian Renner, and Andreas Timm-Giel. 2017. Radio altimeter interference mitigation in wireless avionics intra-communication networks. In *Proceedings of the 15th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*. IEEE, 1–8.

[15] Jeongyoon Heo, Jung-Jun Kim, Saewoong Bahk, and Jeongyeup Paek. 2017. Dodge-jam: Anti-jamming technique for low-power and lossy wireless networks. In *Proceedings of the 14th IEEE International Conference on Sensing, Communication, and Networking*. IEEE, 1–9.

[16] Ye Hu, Nof Abuzainab, and Walid Saad. 2018. Dynamic psychological game for adversarial internet of battlefield things systems. In *Proceedings of IEEE International Conference on Communications*. IEEE, 1–6.

[17] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of International Conference on Learning Representations*.

[18] Kai Lei, Meng Qin, Bo Bai, Gong Zhang, and Min Yang. 2019. GCN-GAN: A non-linear temporal link prediction model for weighted dynamic networks. In *Proceedings of IEEE International Conference on Computer Communications*. IEEE, 1–9.

[19] Chongxuan Li, Kun Xu, Jun Zhu, and Bo Zhang. 2017. Triple generative adversarial nets. In *Advances in Neural Information Processing Systems*, 1–11.

[20] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, Zhen Wang, and Stephen Paul Smolley. 2017. Least squares generative adversarial networks. In *Proceedings of IEEE International Conference on Computer Vision*. IEEE, 2813–2821.

[21] David W. Matolak and Ruoyu Sun. 2017. Air-ground channel characterization for unmanned aircraft systems–part III: The suburban and near-urban environments. *IEEE Transactions on Vehicular Technology* 66, 8 (2017), 6607–6618.

[22] Matthew B. A. McDermott, Tom Yan, Tristan Naumann, Nathan Hunt, Harini Suresh, Peter Szolovits, and Marzyeh Ghassemi. 2018. Semi-supervised biomedical translation with cycle wasserstein regression GANs. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence*, 2363–2370.

[23] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2018. Machine learning with membership privacy using adversarial regularization. In *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*. ACM, 634–646.

[24] Vu Nguyen, Tomas F. Yago Vicente, Maozheng Zhao, Minh Hoai, and Dimitris Samaras. 2017. Shadow detection with conditional generative adversarial networks. In *Proceedings of IEEE International Conference on Computer Vision*. IEEE, 4520–4528.

[25] Tu Dinh Nguyen, Trung Le, Hung Vu, and Dinh Phung. 2017. Dual discriminator generative adversarial nets. In *Advances in Neural Information Processing Systems*, 1–11.

[26] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, 1–9.

[27] Akash Srivastava, Lazar Valkov, Chris Russell, Michael U. Gutmann, and Charles Sutton. 2017. VEEGAN: Reducing mode collapse in GANs using implicit variational learning. In *Advances in Neural Information Processing Systems*, 1–11.

[28] Fengxiao Tang, Zubair Md. Fadlullah, Nei Kato, Fumie Ono, and Ryu Miura. 2018. AC-POCA: Anticoordination game based partially overlapping channels assignment in combined UAV and D2D-based networks. *IEEE Transactions on Vehicular Technology* 67, 2 (2018), 1672–1683.

[29] Jingjing Wang, Chunxiao Jiang, Zhu Han, Yong Ren, Robert G. Maunder, and Lajos Hanzo. 2017. Taking drones to the next level: Cooperative distributed unmanned-aerial-vehicular networks for small and mini drones. *IEEE Vehicular Technology Magazine* 12, 3 (2017), 73–82.

[30] Chaowei Xiao, Bo Li, Jun-Yan Zhu, Warren He, Mingyan Liu, and Dawn Song. 2018. Generating adversarial examples with adversarial networks. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 3905–3911.

[31] Liang Xiao, Caixia Xie, Minghui Min, and Weihua Zhuang. 2018. User-centric view of unmanned aerial vehicle transmission against smart attacks. *IEEE Transactions on Vehicular Technology* 67(4): 3420–3430.

[32] Bo Yang and Min Liu. 2018. Keeping in touch with collaborative UAVs: A deep reinforcement learning approach. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 562–568.

[33] Tianqi Yu, Xianbin Wang, Jiong Jin, and Kenneth McIsaac. 2018. Cloud-orchestrated physical topology discovery of large-scale IoT systems using UAVs. *IEEE Transactions on Industrial Informatics* 14, 5 (2018), 2261–2270.

[34] Chaoyun Zhang, Xi Ouyang, and Paul Patras. 2017. ZipNet-GAN: Inferring fine-grained mobile traffic patterns via a generative adversarial neural network. In *Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies*, 363–375.

[35] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of IEEE International Conference on Computer Vision*. IEEE, 2242–2251.