A GAN-based Covert Channel over Autoencoder Wireless Communication Systems

Ali Mohammadi Teshnizi
Dept. Computer Science
University of Calgary
ali.mohammaditeshniz@ucalgary.ca

Majid Ghaderi
Dept. Computer Science
University of Calgary
mghaderi@ucalgary.ca

Dennis Goeckel
Dept. Electrical and Computer Engineering
University of Massachusetts Amherst
goeckel@ecs.umass.edu

***Abstract*—Covert communication is referred to as a form of communication channel between two parties of Alice and Bob who want to transfer their messages secretly by hiding the presence of their transmissions from a watchful warden Willie. Many works has strived to address the covert channel problem, however, preserving both the signals' statistical properties and the normal communication performance has scarcely been investigated. In this work, we propose a novel covert communication scheme that can be added to any normal autoencoder-based communication without impacting the performance or the statistics of the transmitted signals. We define the covert problem as an optimization problem wherein three covert actors of Alice, Bob, and Willie are represented by a generator, a decoder, and a discriminator neural network. They are jointly trained in an adversarial setting to establish a covert channel in a form of covert noise signals that have the same statistical properties as of the channel's noise regardless of channel type. It is also ensured that our added covert noise signals have the lowest impact on the existing normal communication's error rate. Our results show that our learning-based covert scheme is able to successfully establish a reliable undetectable covert channel between Alice and Bob independent of channel model and cover signals and causes almost no disturbance on the ongoing normal communication of the system.**

I. Introduction

Due to the shared and broadcast nature of wireless channels, there is considerable attention on the security and privacy aspects of wireless communications. While traditional cryptography methods and physical-layer securities can protect the confidentiality of the content (i.e the information transmitted over the channel), there are occasions that hiding the very existence of the communication channel is more vital than securing the communicated message itself. Examples of such situations are military operations, cyber-espionage, social unrest, or communication parties' privacy. All above have motivated the study of hidden communication channels, namely, "covert channels" [1].

The preliminary attempt to obtain covertness started with the study of spread spectrum beginning almost a century ago with the main purpose of hiding military communications. The idea was to spread the transmit power into the noise so that the transmissions are mixed within the noise. Many works continued to further examine different aspects of this idea, however, the fundamental performance limits of such work were unknown until recently when Bash et al. [2] established a square root limit on the number of covert bits that can be reliably sent over an additive white Gaussian noise (AWGN) channel while staying covert to a channel's observer. Followed by this work, there has been a surge of interest in examining covert channels [3], [4] especially in point-to-point wireless communication models.

Numerous works have studied the theoretical limits of covert communication over wireless channels in different scenarios [2], [4]–[6], but only few works have focused on a practical implementation of such channels. In a typical covert communication scenario, there is a specific element of the communication system that Alice and Bob leverage to build their covert communication upon. Some of which to mention are: hardware impairments [7], channel's noise effect [4], presence of a cooperative jammer [3]. Beyond that, the majority of works are based on favorable assumptions to covert users, such as existing a shared secret key between Alice and Bob unknown to Willie, accessibility of covert users to cover signals and modulation type, uncertainty in the knowledge of noise power at the Willie's receiver, neglecting the impact of covert system on the normal communication, and limiting the channel model to AWGN; all of which have led to impracticability of such methods in real world scenarios. Although there is no argue that these works have successfully demonstrated a novel and an unique method of covert communication, but it is always encouraged to study methods that are free as much as viable of these assumptions.

Furthermore, it is important to study applicability of covert channels on next generation communication systems. Similar to many other areas, machine learning (ML) has now found it way to many wireless communication domains [10]. In fact, various network optimization problems, which were traditionally used to be handled with statistical models, are now leveraging on machine learning techniques. Deep neural networks (DNNs) in particular, a major force in machine learning, have answered several wireless problems, such as signal classification, channel estimation, transmitter identification, jamming and anti-jamming [8]. Very recently, as a replacement for conventional modular-based designs, an end-to-end communication model based on deep learning methods has emerged [9]. In this new paradigm, the transmitter and receiver are jointly trained as the encoder and decoder of an autoencoder network. One noticeable difference between end-to-end systems and conventional modular designs is that in

end-to-end systems, the encoder and decoder learn the coding and the modulation tasks simultaneously as opposed to having separate modules for each.

In this work, we introduce a novel deep learning based covert communication method that can be augmented to any autoencoder-based wireless communication and establish a reliable covert channel on top of the existing normal communication. Our proposed covert channel works independent of the channel model, requires no knowledge of cover signals or the modulation type, and it is optimized to have the minimum impact on the communication between normal users while being undetectable to a watchful channel's observer. It is worthwhile to mention that even though we are proposing our method for autoencoder-based wireless communication systems, there is no limitation on integrating our model into existing conventional wireless communication systems.

The contributions of this work can be expressed as:

1) **Cover-Agnostic**: We propose a novel covert communication method using GANs that is independent of cover signals, waveforms, and modulation type of a wireless system.
2) **Channel-Independent**: In our proposed covert scheme, covert users do not have any information about the channel model nor the noise level.
3) **Undetectability**: By forming an adversarial competition between observer and covert users wherein one competes to outperform the other, we ensure the undetectability of covert signals to a high extent.
4) **Low Interference**: We propose a covert communication model that can be integerated into any normal communication system while having almost no disturbance on the ongoing communication between normal parties.

II. Related Works

The main idea of our work stems from stenography techniques, which are commonly used in image steganography. Hence, we first briefly talk about the history and current state of this field of research and then we continue reviewing some of the existing approaches of establishing covert communication at physical layer in a wireless network.

Image Steganography: Deep learning algorithms have proved their efficiency in many aspects. Steganography is one of these areas that has benefited tremendously from deep learning advancements in recent years. The earliest use cases were in steganalysis research. Convolution neural networks (CNNs) for instance, which are generally used in computer vision tasks, showed outstanding results in image steganalysis [11]–[13]. One of the earliest works on image steganography using deep neural networks is a work by [14]. In this work, Baluja proposes a hiding scheme in which the three networks of preparation, hiding, and reveal sort out the secret encoding and decoding task. The disadvantage of these schemes, however, was that the encoding process is reliant on the cover image. To address this, Zhang et al. [16] propose a new architecture in which secret message can be encoded independent of cover images. To manifest robustness against steganalysis practices, researchers started to adopt GAN architectures [17]. Hayes et al. [19] introduce a GAN-based steganography technique that has a different objective for the generator network. Instead of generating cover images, the generator directly learns to embed secret messages into cover images so that the discriminator cannot find the differences. Although this adversarial scenario was preliminary introduced for hiding data in images, researchers found it so versatile that it has now been applied into other forms of data such as video, audio and recently wireless signals.

Covert Communication: One of the first attempts to implement a real-world covert communication is the work by Dutta et al. [20]. They leverage the communication noise caused by either the channel or the hardware imperfections to establish a covert channel. In their proposed method, messages are covertly encoded in the constellation error of normal cover signals. Similarly, Cao et al. [21] further improve this method with the goal of reducing the probability of detection. Hou et al. [22] propose an amplitude based covert channel over LoRa PHY. In their scheme, covert information is embedded with a modulation scheme orthogonal to chirp spread spectrum (CSS). The disadvantage of all the above methods is that they cause distortions in the statistical properties of the system, thus can be detected by an observer. More recent works have explored the viability of deep neural networks in covert communication problem. Sankhe et al. [24] propose a method called Impairment Shift Keying that produces subtle variations in normal signals that can be decoded by CNNs. To achieve the highest covert rate while minimizing the probability of detection, Liao et al. [25] employ a GAN model that can adaptively adjust the signal power at the covert sender. Motivated by the GAN-based steganography technique, Mohammed et al. [7] formulate the covert communication as a three-player game in which three networks are jointly trained. In this setup, the encoder and decoder networks learn to covertly communicate through a form of noise and simultaneously try to confuse a detector network that is responsible for differentiating the expected noise of the system from the added covert perturbations.

While our proposed method shares the same idea as of aforementioned work, there are a couple of deficiencies that our work aims to address. First, our model is not leveraging on any hardware impairment noise and embeds the covert signals into the existing channel's noise. Second, our covert model is completely independent of cover signal whereas covert users of previous work are dependent on the knowledge of cover signals. Third, in the previous work, the impact of added covert signals to the communication in unknown, whereas our model is optimized to preserve the performance of normal communication. Finally yet importantly, previous work assumes the channel between users to be AWGN to simplify the problem, however, in a real wireless communication system, AWGN is not an accurate model to simulate the channel effects since signals are also subject to fading. Our work addresses this by considering two other channel models of Rayleigh and Rician fading.

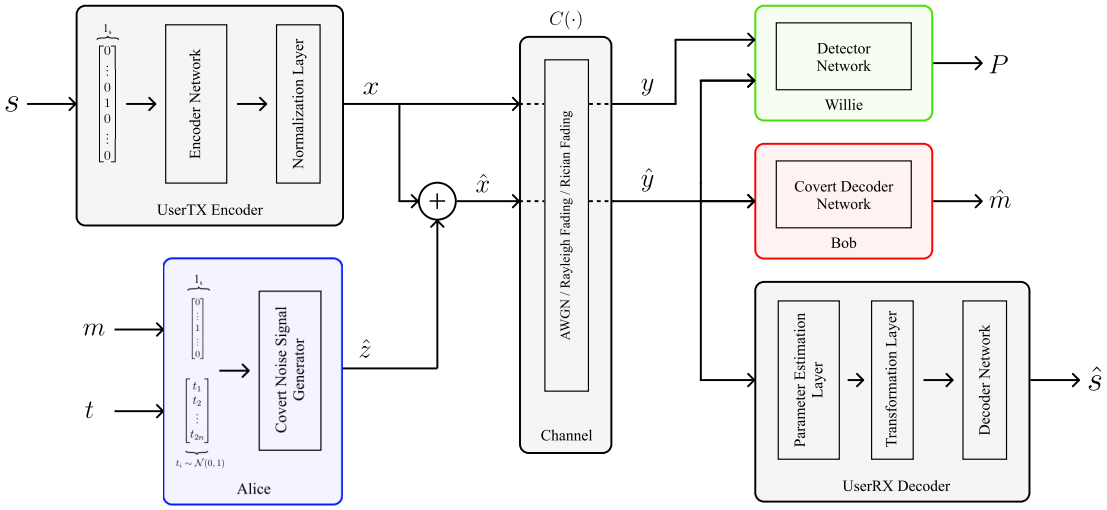


Fig. 1: Overall architecture of our system model. Alice communicating to Bob by generating covert noise signals without disturbing the normal communication between UserTX and UserRX; meanwhile Willie tries to distinguish covert and normal signals (Components under the control of covert users are colored and gray components are the persistent existing parts of the system).

III. System Model

In this section, we give an overall view of the system and define each actor's rule in our model. We also provide a brief background on autoencoder systems.

An overview of our system architecture is shown in figure (1). The system consists of two normal communication parties: normal sender (UserTX), normal receiver (UserRX) who are using an autoencoder model to communicate. The main objective of our work is to establish a covert channel on top of this normal communication without disturbing it. In this model, UserTX uses an encoder network to encode binary messages to a vector of signals. This vector of signals is then transmitted over the channel. We consider three channel models of AWGN, Rayleigh fading, and Rician fading. Signals passing through the channel get distorted by the channel's effect and a noisy version of them are received at the UserRX receiver. Eventually, using a decoder network, UserRX tries to extract the message from the received signals. The remaining three entities of our system are our covert actors. The objective of covert sender (Alice) is to secretly communicate with covert receiver (Bob) by embedding messages in form of perturbations that have similar statistical properties as of the channel's noise. Similar to any covert communication scheme, there is an observer (Willie) whose objective is to monitor the communication and recognize if any covert communication is taking place. All these three covert actors are collaborating and are represented by DNNs. Alice uses a generative model that embeds a confidential message into a covert noise vector. The produced covert signal has to have the lowest impact possible on the normal communication of the system and have enough power to be successfully decoded at Bob's receiver at the same time. This signal is then transmitted over the channel after being added to a normal signal. No matter the channel model, the procedure by which Alice and Bob establish a covert channel stays the same. Observer or warden of the system is responsible for detecting and mitigating any probable covert channel or abnormal communication between entities. Any deviation in the statistical properties of the channel would raise an alarm for the observer. Therefore, Alice and Bob incorporate a statistical undetectability constraint on the produced covert signals. This is achieved by covert users competing an observer (Willie), who is set to rigorously classify signals as covert and normal. As long as Alice succeeds in deceiving Willie to classify covert signals as normal, it is ensured that the added covert noise signals are distributed as the channel's real noise, thus are undetectable.

Background on Autoencoder Systems: An autoencoder-based wireless communication system is an end-to-end learning paradigm that abstracts out the coding and modulation components of a traditional modular communication system by replacing the transmitter and receiver with DNNs. The encoder (transmitter) first uses a mapping to transform k bits of data into a message s where $s \in \{1, ..., M\}$ and $M = 2^k$. Then it takes this transformed message as an input and generates a signal $x = E(s) \in \mathbb{R}^{2n}$, which is a real valued vector. This $2 \times n$ dimensional real valued vector can be treated as an n dimensional complex vector where n is the number of channel uses that are needed for the signal to be transmitted over. Then, the channel's noise effect z, which is usually considered to be AWGN, is added to the signal vector. Thus, the received signal at the receiver, carrying the noise of the channel, can be expressed as $y = x + z$. Rayleigh and Rician fading channels is used in case of existence of objects in the environment subjecting signals to fading. In this channel model, received signal is given by $y = h \cdot x + z$, where $h \sim \mathcal{CN}(0, 1)$ is the fading channel gain. Regardless of channel model, the decoder (receiver) applies the transformation $D : \mathbb{R}^{2n} \rightarrow M$ to outputs the reconstructed version of the message s, which is denoted as $\hat{s} = D(y)$.

IV. Covert Channel Model

For a given binary secret message m, Alice first one-hot encodes the message and then uses its generator model to produce a covert noise signal $\hat{z}$. This covert signal is then added to a vector of normal signal x, which is carrying a message between UserTX and UserRX. Therefore, the covert signals before being transmitted over the channel can be denoted as:

$$\hat{x} = x + \hat{z}. \tag{1}$$

The signal is then transmitted over the channel. We mentioned that we assume the channel between sender and receiver to be AWGN or Rayleigh. Therefore, there will be two different channel outputs for these two different channel models. We express the distortions caused by the channel as a mapping function $C(\cdot)$.

AWGN Channel Output: For the AWGN channel model, the signal received at the receiver carries within itself the channel noise effect $z \sim \mathcal{N}(0, \sigma^2_{chl})$. Thus, the channel function $C(\cdot)$ and final covert signal $\hat{y}$ can be represented as:

$$\hat{y} = C(\hat{x}) = \hat{x} + z. \tag{2}$$

Rayleigh and Rician Fading Channel Output: For the Rayleigh and Rician fading channel models, we consider a flat block fading channel where each codeword is assumed to be faded independently. Let h be the fading coefficient for transmitting the codeword $\hat{x}$, then the channel function $C(.)$ and and the final covert signal $\hat{y}$ is given by:

$$\hat{y} = C(\hat{x}) = h \cdot \hat{x} + z. \tag{3}$$

On the receiver side, Bob receives a transmitted signal $\hat{y}$ after the channel noise being added into it. He uses its decoder network to reconstruct the covert message $\hat{m}$. Meanwhile, the UserRX uses the same signal to extract the normal message $\hat{s}$, which is the reconstructed message of s.

The statistical properties of signals transmitted over the channel are captured by Willie. His objective is to classify sequences of normal y and covert signals $\hat{y}$ and provide useful feedback to Alice. This feedback helps Alice to modify the produced covert signals such that they are indistinguishable from normal transmitted signals. In other words, it ensures that both normal and covert signals have similar statistical properties.

A. General Formulation

The very first objective of our covert model is to have a working covert channel. To this end, Bob has to have a plausible accuracy in decoding covert messages that Alice sends through the covert signals $\hat{y}$. As mentioned in previous section, Alice employs a generative model instead of an encoder model suggested by [7]. Using an encoder model to produce covert signal perturbations will map each covert message m to a single covert noise vector $\hat{z}$. Inevitably, these deterministic covert perturbations can be detected and averaged out with ease by a careful observer or a defender as already shown in a work of Bahramali et al. [8] studying a relatable covert attack problem against autoencoder wireless networks. Thus, we use an stochastic generative model for Alice so that each covert message gets mapped to a set of different covert noise signals. Let $A(\cdot)$ be the underlying function of Alice's generative model that takes a random trigger $t \sim \mathcal{N}(0, 1)$ and a covert message m and produces a covert signal $\hat{z}$ (the corresponding covert signal then can be denoted as $\hat{z}_{m,t} = A(m, t)$). Let also $B(\cdot)$ be the underlying function of the decoder network that Bob makes use of to reconstruct the covert message $\hat{m}$. Then the reliability of communication between Alice and Bob is achieved using the below loss function:

$$\begin{aligned} \mathcal{L}_{Bob} &= \mathbb{E}_m[H(\hat{m}, m)] \\ &= \mathbb{E}_m[H(B(\hat{y}), m)] \\ &= \mathbb{E}_m[H(B(C(\hat{x}), m)] \\ &= \mathbb{E}_m[H(B(C(A(m, t) + x)), m)]. \end{aligned} \tag{4}$$

where $H(\cdot)$ is the cross entropy between the probability of reconstructed covert message $\hat{m}$ and the actual covert message m. This equation can be used for optimizing both Alice's and Bob's networks by freezing one or the other network's parameters iteratively. While (4) ensures the communication accuracy, we also need to consider that the generated perturbations should leave no detrimental impact on the normal communication between UserTX and UserRX, otherwise causing an unexpected increase in the error rate of communication is deemed as an abnormal behavior by Willie. We apply this constraint by minimizing the autoencoder's loss function during Alice's training:

$$\begin{aligned} \mathcal{L}_{UserRX} &= \mathbb{E}_m[H(\hat{s}, s)] \\ &= \mathbb{E}_m[H(D(\hat{y}), s)] \\ &= \mathbb{E}_m[H(D(C(\hat{z} + x)), s)] \\ &= \mathbb{E}_m[H(D(C(A(m, t) + E(s))), s)]. \end{aligned} \tag{5}$$

where $D(\cdot)$ is UserRX's decoder network function, and $E(\cdot)$ is the underlying function of the UserTX's encoder network. Note that both UserTX's encoder and UserRX's decoder networks are frozen during this training and only Alice's

Algorithm 1 Optimizing covert models algorithm

$X \leftarrow$ normal signals data
$S, M \leftarrow$ normal and covert messages sets
$A, B, W \leftarrow$ Alice, Bob, and Willie network functions
$D \leftarrow$ UserRX decoder network function
$\mathcal{H} \leftarrow$ cross entropy function
$C \leftarrow$ channel's remapping function
for epoch $ep \in \{1 \dots n_{epochs}\}$ **do**
  $t \sim \mathcal{N}(0, 1)$
  $\mathcal{L}_{Willie} = H(C(X), C(A(M, t) + X))$
  Update W to minimize $\mathcal{L}_{Willie}$
  $\mathcal{L}_{Bob} = H(C(A(M, t) + X), M)$
  Update B to minimize $\mathcal{L}_{Bob}$
  $\mathcal{L}_{UserRX} \leftarrow H(D(C(A(M, t) + X)), S)$
  $\mathcal{L}_{Alice} = \lambda_{Bob}\mathcal{L}_{Bob} + \lambda_{UserRX}\mathcal{L}_{UserRX} - \lambda_{Willie}\mathcal{L}_{Willie}$
  Update A to minimize $\mathcal{L}_{Alice}$
end for

parameters are updated.

In our model, the observer entity or Willie, acts as the discriminator in GAN models [17]. The so-called real and fake samples in GANs' discriminator network training process is here mapped to normal and covert signals, respectively. Thus, we express the loss function of Willie as:

$$\begin{aligned}\mathcal{L}_{Willie} &= \mathbb{E}_m[H(\hat{y}, y)] \\ &= \mathbb{E}_m[H(C(\hat{x}), C(x))] \\ &= \mathbb{E}_m[H(C(A(m,t)+x), C(x))].\end{aligned} \tag{6}$$

where $H(\cdot)$ here is the binary cross entropy between the covert signal $\hat{y}$ and the normal signal y. This white-box adversarial training against Alice's network ensures that Willie will be adequately trained to tell covert and non-covert signals apart. On the other hand, we do not want the covert signals that Alice produces to deviate from the statistical properties of the normal signals on the channel, otherwise it is likely that the observer of the channel detects and mitigates the covert communication. To achieve this undetectability property, we pose a new constraint on Alice's optimization function for maximizing Willie's uncertainty about his predictions. Having a regularizer as such helps Alice and Bob to form their covert communication in a way that is indistinguishable from the actual channel's noise, yet understandable by both. Altogether, Alice's loss function can be expressed as a weighted sum of three different objectives:

$$\mathcal{L}_{Alice} = \lambda_{Bob}\mathcal{L}_{Bob} + \lambda_{UserRX}\mathcal{L}_{UserRX} - \lambda_{Willie}\mathcal{L}_{Willie}. \tag{7}$$

where λ_{Bob}, λ_{UserRX}, and λ_{Willie} determine the importance of each objective for training Alice's network. Algorithm 1 summarizes the procedure by which we train our covert models.

B. Neural Network Architecture

Before discussing the architecture of our neural network models, we need to state the focus of this work is not to introduce an autoencoder wireless network, so we only give a brief description on how this model works. A more detailed explanation of such a network can be found in the original paper [9].

Autoencoder's Network: As proposed in the original autoencoder wireless communication paper, autoencoder model accepts a binary message s of size k bits and outputs a reconstructed version of it $\hat{s}$. Figure 2 depicts the overall architecture of our autoencoder model. The encoder part of the model first one-hot encodes the message and then maps it to a vector of signals of size $2 \times n$, where n is the number of channel uses. This transmitted signal is then given to a mapping function that applies the channel effects. On the receiver side, two layers of parameter estimation and transformation only becomes active if channel model is Rayleigh or Rician fading. In our model, transformation function is a simple division function that divides the received signal by the estimated channel fading coefficients by the parameter estimation. Note that more complex transformation functions

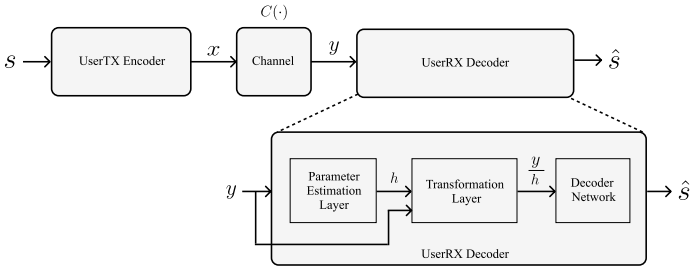
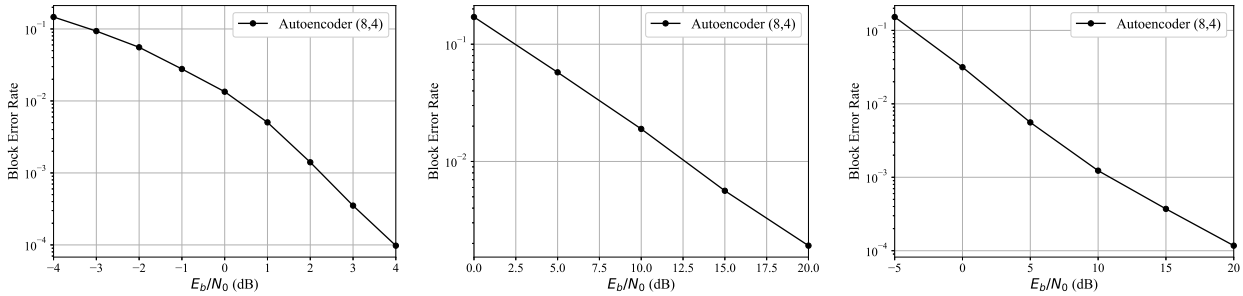


Fig. 2: Detailed architecture of UserRX's decoder network. Data passes through parameter estimation and transformation layers only when channel is Rayleigh or Rician.

can be used and are described in [9], however optimizing the performance of autoencoder model is out of the scope of this article. Eventually, the transformed signal is fed to the decoder's network and the original normal message is reconstructed.

Alice's Network: Similar to the Autoencoder's encoder network, Alice takes a covert message m and transforms it to its corresponding one-hot encoding representation of it so that each message belongs to a unique class. Next, given a random trigger t, Alice uses its generator model to produce a covert noise signal $\hat{z}$ and then adds it to a normal signal x that is being transmitted at the time. For the Alice's generator model, we use multiple dense layers with ReLU and Tanh activation functions. The first layer of this model takes a trigger number t and an one-hot encoded covert message m, and acts as embedding layer by enlarging the input's domain space. The following fully contacted layers are to extract the useful features and do the encoding process. The last layer of this model does a dimension transformation so that the generated covert signal $\hat{z}$ complies with the dimension of the normal signal x on the channel.

Bob's Network: Bob receives this covert signal $\hat{y}$ that has undergone the channel's effects and feeds it through its decoder network regardless of what the channel model is and extracts the secret message by doing classification on the signal. Bob's network has a more complicated structure comparing to Alice as it has to decode the secret message from a signal $\hat{y}$ that has been distorted stiffly as a result of going through the channel. The received message by Bob first goes through the first layer of the network, which is a wide dense layer with a Tanh activation function, to increase the input's feature space. Then the data is passed through multiple 1-Dimensional Convolutional (1D Conv) layers that supposedly learn the coding that Alice has fabricated to encode the covert messages. We have found that using 1D Conv layers helps Bob and Alice achieving a better consistency in the accuracy of their communication, especially when the channel model is more complicated (i.e. when there is also fading in the channel). The rest of Bob's decoder network consists of two dense layers that does a domain remapping from the learned feature space to the covert message domain space. Similar to the UserRX's decoder network, Bob eventually predicts the covert message by doing a classification on the received signal.



(a) AWGN channel (b) Rayleigh fading channel (c) Rician fading channel

Fig. 3: Trained Autoencoder's BLER over a range of SNR values.

Alice	
Layer	Output dimension
Input (size 8 + 2^k)	-
Dense + ReLU	32 + 2^{k+1}
Dense + ReLU	32 + 2^{k+1}
Dense + ReLU	8 × 2^k
Dense	8 × 2
Bob, Willie	
Input (size 2 × 8) Dense + Tanh	2 × 8
Convolutional (8 filters, kernel size 1 × 1, stride 1) + LeakyReLU	8 × 16
Convolutional (8 filters, kernel size 1 × 2, stride 1) + LeakyReLU	8 × 15
Convolutional (8 filters, kernel size 1 × 4, stride 2) + LeakyReLU	8 × 6
Convolutional (8 filters, kernel size 1 × 2, stride 1) + LeakyReLU	8 × 5
Convolutional (8 filters, kernel size 1 × 2, stride 1) + LeakyReLU	8 × 4
Flatten	32
Dense + Tanh	16
Dense + (Willie: Sigmoid, Bob: Softmax)	Willie: 1, Bob: 2^k

TABLE I: Alice, Bob, and Willie's networks detailed architecture.

Encoder	
Layer	Output dimension
Input (size 16)	-
Dense + ELU	16
Dense + ELU	2 × 8
Convolutional (8 filters, kernel size 1 × 2, stride 1) + Tanh	8 × 15
Convolutional (8 filters, kernel size 1 × 4, stride 2) + Tanh	8 × 6
Convolutional (8 filters, kernel size 1 × 2, stride 1) + Tanh	8 × 5
Convolutional (8 filters, kernel size 1 × 2, stride 1) + Tanh	8 × 4
Flatten	32
Dense	2 × 8
Normalization	2 × 8
Parameter Estimation	
Dense + ELU	2 × 16
Dense + Tanh	2 × 32
Dense + Tanh	2 × 8
Dense	2 × 1
Decoder	
Layer	output dimension
Dense + Tanh	2 × 8
Convolutional (8 filters, kernel size 1 × 2, stride 1) + Tanh	8 × 15
Convolutional (8 filters, kernel size 1 × 4, stride 2) + Tanh	8 × 6
Convolutional (8 filters, kernel size 1 × 2, stride 1) + Tanh	8 × 5
Convolutional (8 filters, kernel size 1 × 2, stride 1) + Tanh	8 × 4
Flatten	32
Dense + Tanh	2 × 8
Dense + Tanh	2 × 8
Dense + Softmax	16

TABLE II: Autoencoder's network detailed architecture.

Willie's Network: Willie receives both the covert signal $\hat{y}$ and the normal signal y and outputs a confidence probability P on how probable it is for the signal to be normal. We choose the same network architecture of Bob's for Willie except for the last layer that has a Sigmoid activation function instead of Softmax. This ensures that Bob and Willie has the same capacity of training and can be compete each other in a fair setup.

V. Experiments and Evaluation

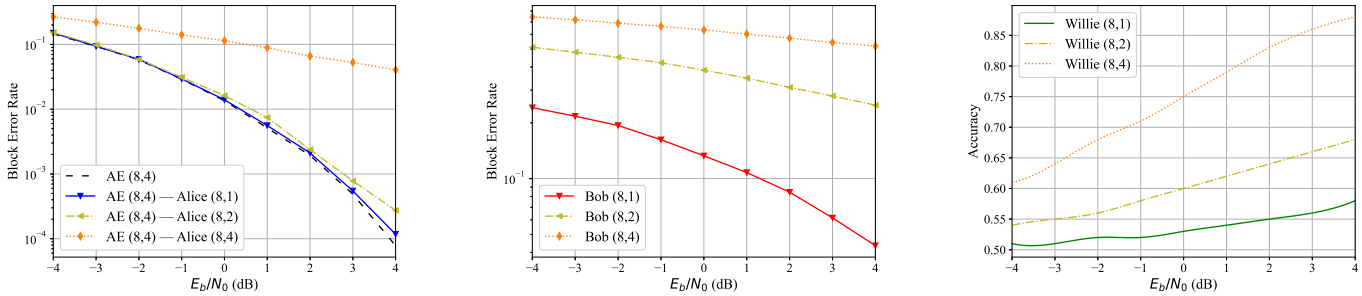
We categorize our experiments into two different sections. In the first section we give performance evaluation information on our trained autoencoder network. Next, we discuss the results for our implemented covert models.

A. Baseline Autoencoder Performance

We implemented an autoencoder communication network for the normal communication between UserRX and UserTX. Based on the notation used in [9], an $Autoencoder(n, k)$ is a neural network communication model that sends k bits of data in n channel uses. We choose these two numbers of channel uses n and the binary message of size k to be 8 and 4, respectively. These numbers are chosen this way so that we could evaluate the performance of our trained autoencoder model with the results given on [9]. Nevertheless, our covert model works independent of these parameters and can be used for any autoencoder communication setup. In order to train our autoencoder model, we generate two datasets of train and test by generating random binary messages s of size k. There are 8192 random binary messages in the training set and 51200 random binary messages in the test set. We intentionally created a much larger data set for testing to make sure that each symbol y undergoes various channel distortions to have an accurate evaluation of the model's performance. We set the learning rate to 0.001 and optimized the model using the Adam optimizer [26]. We choose the batch size to be 64 and train the model for 100 epochs. For the channel configuration, we choose a fixed signal to noise ratio (SNR) value during training. The SNR value for the AWGN channel is set to 4dB, and we give the higher SNR value of 16dB to the Rayleigh fading channel due to the channel complexity. Figure 3 shows the performance of our trained normal communication models in terms of block error rate (BLER) for a range of SNR values under AWGN and Rayleigh fading channel conditions.

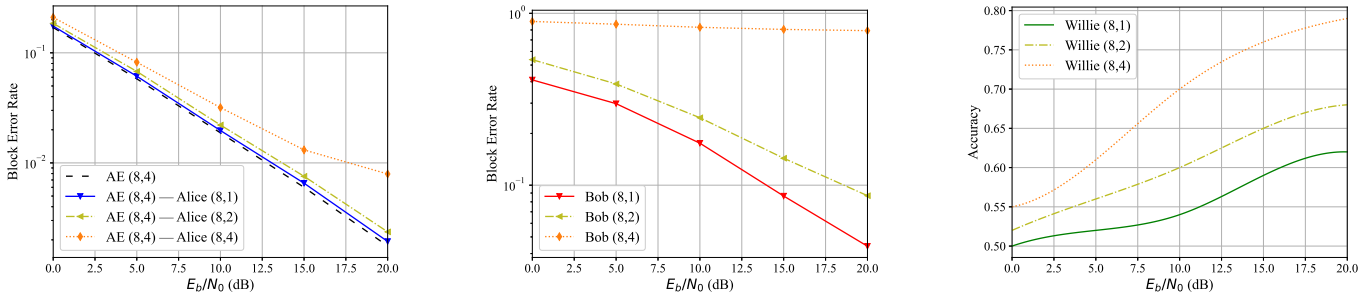
B. Covert Models Evaluation Results

As for the covert models, we evaluate our system's performance on the two different channel models of AWGN and Rayleigh fading. In both settings, we use the same training procedure and network architecture for our covert models. We start our experiment by sending 1 bit of covert data over 8



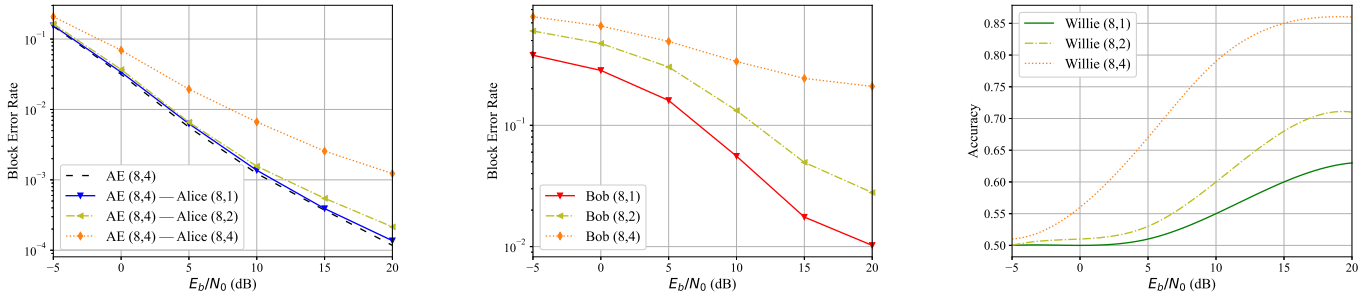
(a) Autoencoder's BLER (b) Bob's BLER (c) Willie's accuracy

Fig. 4: Trained covert models' performance over AWGN channel for different covert data rates on a range of SNR values.



(a) Autoencoder's BLER (b) Bob's BLER (c) Willie's accuracy

Fig. 5: Trained covert models' performance over Rayleigh fading channel for different covert data rates on a range of SNR values.

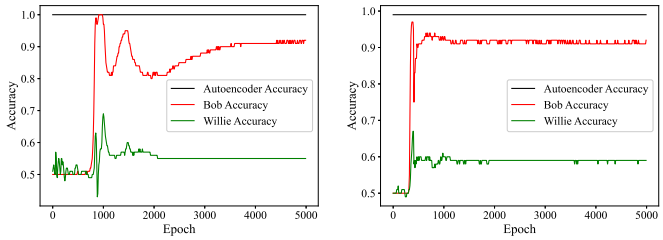


(a) Autoencoder's BLER (b) Bob's BLER (c) Willie's accuracy

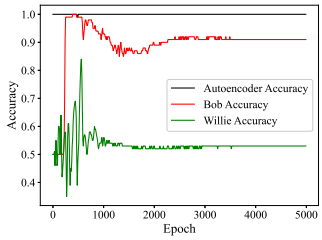
Fig. 6: Trained covert models' performance over Rician fading channel for different covert data rates on a range of SNR values.

channel uses and then gradually increase the number of covert bits to see how increasing the covert data rate will effect each component of our covert scheme. The notations $Alice(n,k)$, $Bob(n,k)$), and $Willlie(n,k)$ are used to differentiate models operating on different bit rates and the interpretation of it is just the same as what was explained for the autoencoder model. Since each covert message has to be paired with a normal message, we generate the train and the test covert messages m set to have the same number of train and test messages as of the autoencoder's. All models are jointly trained for 5000 epochs using Adam optimizer. We adjust the importance of each objective for Alice's training by setting $\lambda_{Willie} = 2\lambda_{Bob} = 4\lambda_{UserRX}$ in (7). We start the training with the learning rate of 0.001 and gradually halve the learning rate after every 500 epochs. In each epoch, we first update parameters of Willie's network using (6), then train Alice's network for one step using (7), and eventually optimize Bob's network based on (4). Although we train our autoencoder network on a fixed SNR value, we find our covert scheme performs better when trained on a range of SNR values. Training our models this way, not only helps Alice to better preserve the normal communication's accuracy but also makes Bob to be able to decode covert messages more accurately on lower SNR values. Accordingly, we set the SNR value to be in the range of -2dB to 8dB for the AWGN channel and 10dB to 20dB for both the Rayleigh and the Rician fading channels.

Training Procedure: Figure 7 shows the progress of each covert actor's accuracy on the test set during the training process. As the training goes on, Bob gradually learns to decode covert messages m and establishes a reliable com-



(a) AWGN channel (b) Rayleigh fading channel



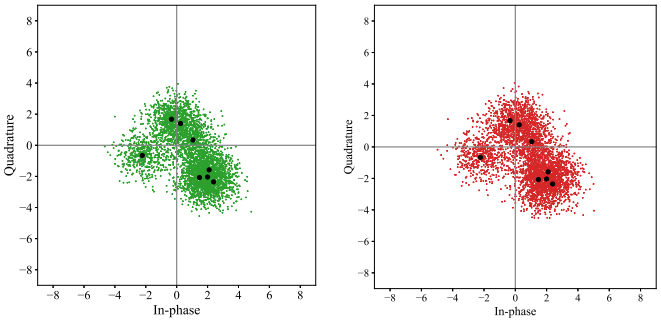
(c) Rician fading channel

Fig. 7: Evaluation results of our covert and autoencoder models during training process show system reaching a stable point after successful training.

munication with Alice. After a few epochs, when the covert communication begins to take up and stabilizes, signals start to deviate from the distribution they had, causing Willie to better detect covert signals. When Willie's accuracy increases, the term $\mathcal{L}_{Willie}$ dominates the other two objectives of Alice's loss function in (7). This causes Alice to gradually sacrifice its accuracy for the sake of undetectability. Soon afterwards, the training process reaches a stable point where neither of covert models sees any noticeable improvement in their accuracy as the training continues.

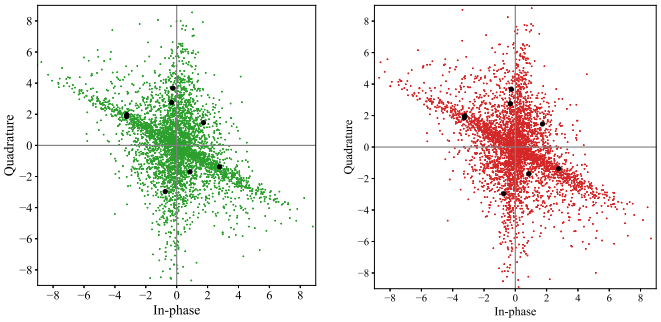
Covert Rate: Figures 4, 5, and 6 show our final results. It also demonstrates how the accuracy of our each covert models changes as we increase the covert rate. As we expected, the covert communication becomes more unreliable, the impact on normal communication increases, and the detection becomes easier on higher covert rates.

Undetectibility: In figures 4(c), 5(c), and 6(c) the accuracy of Willie is shown in percentage over a range of SNR values for flagging signals as covert and normal. These plots gives us some intuition on how probable it will be for our covert signals to be detected by a target detector at each SNR value for different covert rates. Figures 8),9), and 10 compare the constellation cloud of a covert and a normal signal for AWGN, Rayleigh and Rician fading channels. We have marked each symbol of the encoder's ouput signal x as black circle points on the constellation diagrams. Red constellation cloud shows how covert signals scatter after going through the channel and the green cloud shows this for normal signals. Since data is sent over 8 channel uses, there are 8 black points on the chart. To be consistent with Willie's accuracy and Bob's error rate for our both channel models, we have set the SNR value to 2dB for the AWGN and 15dB for the Rayleigh fading channel and 8db for the Rician fading channel so that in all channel models, the probability of detection remains relatively the same and the covert communication BLER stays below 10^{-1}. This area of operation gives Alice and Bob a fair covert communication reliability while maintaining their covertness. As it is also evident in these figures, comparing the signal constellation diagrams before and after our covert model applied shows that Alice has perfectly learned to cloak the covert signals into the distribution of the channel's noise after a successful training procedure.



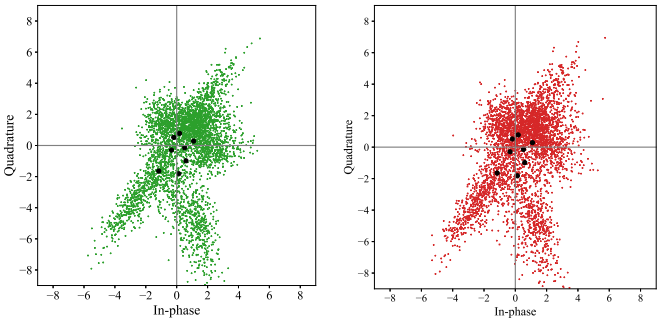
(a) Without covert transmission (b) With covert transmission

Fig. 8: Comparing AWGN channel constellation clouds of a signal before and after our covert scheme being applied.



(a) Without covert transmission (b) With covert transmission

Fig. 9: Comparing Rayleigh fading channel constellation clouds of a signal before and after our covert scheme being applied.



(a) Without covert transmission (b) With covert transmission

Fig. 10: Comparing Rician fading channel constellation clouds of a signal before and after our covert scheme being applied.

VI. Conclusion

In this paper, we introduced a novel deep learning-based covert communication scheme using adversarial training and generative models. We showed how our covert sender establishes a reliable and an undetectable covert channel by hiding the covert messages into the channel's noise by help of an observer network so that the channel's statistical properties remain intact. Our results indicate that our model works does not rely on the channel model and requires no knowledge of cover signals. We also provided information on the performance of covert model on three channel models and noise levels for different covert rates. Furthermore, we investigated the impact of our added covert signals on the performance of the autoencoder-based communication system and verified our covert scheme causes no disturbance on the existing communication between normal users. Our results indicate that our covert models were successfully trained to covertly communicate while having a minimum impact on the normal communication of the system.

References

[1] B. W. Lampson, "A note on the confinement problem," *Communications of the ACM*, vol. 16, no. 10, pp. 613–615, 1973.

[2] B. A. Bash, D. Goeckel, and D. Towsley, "Square root law for communication with low probability of detection on awgn channels," in *2012 IEEE International Symposium on Information Theory Proceedings*. IEEE, 2012, pp. 448–452.

[3] T. V. Sobers, B. A. Bash, S. Guha, D. Towsley, and D. Goeckel, "Covert communication in the presence of an uninformed jammer," *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 6193–6206, 2017.

[4] R. Soltani, D. Goeckel, D. Towsley, B. A. Bash, and S. Guha, "Covert wireless communication with artificial noise generation," *IEEE Transactions on Wireless Communications*, vol. 17, no. 11, pp. 7252–7267, 2018.

[5] A. Sheikholeslami, M. Ghaderi, D. Towsley, B. A. Bash, S. Guha, and D. Goeckel, "Multi-hop routing in covert wireless networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 6, pp. 3656–3669, 2018.

[6] K. Li, M. Ghaderi, and D. Goeckel, "Fundamental limits of activity-based covert channels," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 1–6.

[7] H. Mohammed, X. Wei, and D. Saha, "Adversarial learning for hiding wireless signals," in *2021 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2021, pp. 01–06.

[8] A. Bahramali, M. Nasr, A. Houmansadr, D. Goeckel, and D. Towsley, "Robust adversarial attacks against dnn-based wireless communication systems," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, 2021, pp. 126–140.

[9] T. O'shea and J. Hoydis, "An introduction to deep learning for the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, 2017.

[10] T. Wang, C.-K. Wen, H. Wang, F. Gao, T. Jiang, and S. Jin, "Deep learning for wireless physical layer: Opportunities and challenges," *China Communications*, vol. 14, no. 11, pp. 92–111, 2017.

[11] S. Tan and B. Li, "Stacked convolutional auto-encoders for steganalysis of digital images," in *Signal and information processing association annual summit and conference (APSIPA), 2014 Asia-Pacific*. IEEE, 2014, pp. 1–4.

[12] Y. Qian, J. Dong, W. Wang, and T. Tan, "Deep learning for steganalysis via convolutional neural networks," in *Media Watermarking, Security, and Forensics 2015*, vol. 9409. SPIE, 2015, pp. 171–180.

[13] G. Xu, H.-Z. Wu, and Y.-Q. Shi, "Structural design of convolutional neural networks for steganalysis," *IEEE Signal Processing Letters*, vol. 23, no. 5, pp. 708–712, 2016.

[14] S. Baluja, "Hiding images in plain sight: Deep steganography," *Advances in neural information processing systems*, vol. 30, 2017.

[15] C. Zhang, C. Lin, P. Benz, K. Chen, W. Zhang, and I. S. Kweon, "A brief survey on deep learning based data hiding, steganography and watermarking," *arXiv preprint arXiv:2103.01607*, 2021.

[16] C. Zhang, P. Benz, A. Karjauv, G. Sun, and I. S. Kweon, "Udh: Universal deep hiding for steganography, watermarking, and light field messaging," *Advances in Neural Information Processing Systems*, vol. 33, pp. 10 223–10 234, 2020.

[17] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[18] D. Volkhonskiy, I. Nazarov, and E. Burnaev, "Steganographic generative adversarial networks," in *Twelfth International Conference on Machine Vision (ICMV 2019)*, vol. 11433. International Society for Optics and Photonics, 2020, p. 114333M.

[19] J. Hayes and G. Danezis, "Generating steganographic images via adversarial training," *Advances in neural information processing systems*, vol. 30, 2017.

[20] A. Dutta, D. Saha, D. Grunwald, and D. Sicker, "Secret agent radio: Covert communication through dirty constellations," in *International Workshop on Information Hiding*. Springer, 2012, pp. 160–175.

[21] P. Cao, W. Liu, G. Liu, X. Ji, J. Zhai, and Y. Dai, "A wireless covert channel based on constellation shaping modulation," *Security and Communication Networks*, vol. 2018, 2018.

[22] N. Hou and Y. Zheng, "Cloaklora: A covert channel over lora phy," in *2020 IEEE 28th International Conference on Network Protocols (ICNP)*. IEEE, 2020, pp. 1–11.

[23] L. Bonati, S. D'Oro, F. Restuccia, S. Basagni, and T. Melodia, "Stealte: Private 5g cellular connectivity as a service with full-stack wireless steganography," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.

[24] K. Sankhe, F. Restuccia, S. D'Oro, T. Jian, Z. Wang, A. Al-Shawabka, J. Dy, T. Melodia, S. Ioannidis, and K. Chowdhury, "Impairment shift keying: Covert signaling by deep learning of controlled radio imperfections," in *MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)*. IEEE, 2019, pp. 598–603.

[25] X. Liao, J. Si, J. Shi, Z. Li, and H. Ding, "Generative adversarial network assisted power allocation for cooperative cognitive covert communication system," *IEEE Communications Letters*, vol. 24, no. 7, pp. 1463–1467, 2020.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.