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. Although there exists a huge body of work on covert communication examining multiple aspects of this topic, there is a lack of studies investigating viability of having such channels on recently introduced autoencoder-based wireless systems. In this work, we propose a novel deep learning based covert communication scheme that runs on top of an autoencoder communication system. 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 jointly trained in an adversarial setting. The objective is to establish a covert channel in a form of covert noise signals that have the same statistical properties as of the channel's noise. Additionally, we ensure 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 successfully able to establish a reliable undetectable channel between Alice and Bob while causing almost no disturbance on the ongoing 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 began almost a century ago with the main purpose of hiding military communications. The idea is 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.

In the last decade, the majority of research works on covert communications focus on integration of covert communication techniques into the current and future wireless technologies, such as 6G wireless networks and IoT. With the advent of AI and recent advances in computer's computation power and algorithmic designs, machine learning (ML) is now becoming an integral part of many research topics. Correspondingly, wireless community started to adopt machine learning techniques and use them as solutions to the various network optimization problems, which were traditionally used to be handled with statistical models. Deep neural networks (DNNs) in particular, a major force in machine learning, are now applied into many of these tasks including signal classification, channel estimation, transmitter identification, jamming and anti-jamming [25].

More recently, as a replacement for conventional modular-based designs, an end-to-end communication model based on deep learning methods has emerged [5]. In this new paradigm, the transmitter and receiver are jointly trained as the encoder and decoder of an autoencoder network. While the transmitter learns to encode a symbol to an embedding vector of wireless signals, the decoder, on the other side, learns to retrieve that symbol back from the transmitted signal that has gone through the channel. One noticeable difference between end-to-end systems and conventional modular designs is that in end-to-end systems, the encoder learns coding and modulation tasks simultaneously as opposed to having separate modules to perform each task. This is also true for the decoder of the system that jointly learns demodulation and decoding. Given the fact that channel distortions usually have a non-linear behavior, a DNN-based communication model is expected to capture these dissimilitudes better compared to a statistical model. That means, these systems can learn more complex channel effects and can operate more robustly under different noise levels and channel imperfections [6]. In this work, we introduce a novel deep learning based covert communication method inspired by steganography techniques using generative adversarial networks (GANs). The proposed covert scheme is devised for the next generation autoencoder-

based wireless communication models, which are believed to replace traditional modular based wireless systems in the near future. However, there is no limitation on integrating our covert model to the current wireless communication systems as well. The contributions of this work can be expressed as:

1) We propose a novel covert communication method using GANs that works independent of the channel and cover signals of the wireless system.
2) We train and evaluate our system on two different channel models of AWGN and Rayleigh fading to be compliant with a realistic wireless communication scenario.
3) We also measured both the performance of our covert model and the normal autoencoder-based communication system in terms of error rate to comprehend both the effectiveness of covert scheme and the impact of it on the existing communication between normal users.

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.

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 [7]–[9], replacing the traditional statistical methods. One of the earliest works on image steganography using deep neural networks is a work by [10]. 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 preparation network transforms the hidden message into features that are commonly used for compressing images. Then, the hiding network embeds it into a cover image and sends it to the reveal network, where the secret message gets extracted from the cover image. Followed by this work, researchers discovered that the existence of preparation network is not necessary and the framework can be expressed in a simpler form by excluding this network from the model [11]. The disadvantage of these schemes, however, was that the encoding process is reliant on the cover image. To address this, Zhang et al. [12] propose a new architecture in which secret message can be encoded independent of cover images. Beside having more flexibility on hiding the information, this approach has also become an effective method for image watermarking. To manifest robustness against steganalysis practices, researchers started to adopt GAN architectures. In a typical adversarial network setting there are two neural networks involved, a generator network (G) and a discriminator network (D). These two networks are then trained against each other where the generator tries to deceive the discriminator by generating data similar to those of in the training set and the discriminator tries to correctly categorize them as fake and real [13]. Volkhonskiy et al. [14] propose a steganography technique based on GAN training. The main idea of their work is to use a generative network to produce a new set of cover images that when carry the secret message using any of the available steganography techniques are less exposed to be detected by a discriminator network (i.e. a steganalysis network). Similarly, Hayes et al. [15] 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.

Numerous works have studied the theoretical limits of covert communication over wireless channels in different scenarios [2], [4], [16], [17], but only few works have focused on a practical implementation of such channels. One real world example of a covert communication is the work by Dutta et al. [18]. 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. [19] further improve this method with the goal of reducing the probability of detection. Hou et al. [20] 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). Bonati et al. [21] introduce SteaLTE which is a full-stack wireless steganography method on softwarized cellular networks. To covertly modulate symbols, they employ the three different approaches of dirty QPSK modulation, hierarchical amplitude shift keying (ASK) manipulation, and phase offset of the primary symbols modification. The disadvantage of all the above methods is that the distortions caused in the statistical properties of the system can be detected with high confidence by a careful observer. More recent works have explored the viability of deep neural networks in covert communication problem. Sankhe et al. [22] propose a method called Impairment Shift Keying that produces subtle variations in normal signals in a controlled way such that a CNN model can be trained to classify them as zeros or ones. To achieve the highest covert rate while minimizing the probability of detection, Liao et al. [23] 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. [24] 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 this work, there are a couple of deficiencies in the previous work that ours aims to address. First, we study covert communication over Autoencoder-based wireless systems which will soon replace the traditional modular systems in future wireless communication technologies such as 6G. Second, the authors of previous work state that their covert scheme is independent of cover signals, however, they assume the modulation type is known to the covert receiver; thus, the covert receiver knows what the cover signal is and only by subtracting it from the received signal is that the receiver can recover the covert message. In our work, however, covert receiver extracts the secret message independent of this knowledge. Third, the previous work evaluates the performance of their proposed covert communication model but gives no information on the impact of their added noise on the normal communication of the system. It is of utmost importance that the added covert noise signals do not interfere with the normal communication of the system, since any unexpected increase in the communication's error rate would be an indication of a suspicious activity (i.e. in this case, a covert communication) taking place. Finally yet most 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 also considering Rayleigh fading as the communication channel.

III. Background On Autoencoder Wireless 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 need 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 fading channel is also used when there are many objects in the environment and signals become subject 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. Our Covert Communication System Model

In this section, we begin with an overview of our covert communication model and will further discuss the details of our scheme in the subsequent sections.

A. Overview

An overview of our system architecture is shown in figure (1). The main idea of our work is to establish a covert channel on top of a normal communication between a sender (UserTX) and a receiver (UserRX) who are using an autoencoder-based wireless system for their communications. We consider both an AWGN and a Rayleigh fading channel between our communication parties. The objective of the covert sender (Alice) is to secretly communicate with the covert receiver (Bob) by embedding messages in form of perturbations that have similar statistical properties as of the channel's noise. This is achieved by help of an observer (Willie) who tries to rigorously classify transmitted signals as covert and non-covert after they pass through the channel.

B. System Architecture

As mentioned above, there are three main actors in our covert system which we call them by their placeholder names: Alice, Bob, and Willie. All three are collaborating in our covert scheme and are represented by DNNs. Alice uses a generative model that embeds a confidential message m into a covert noise vector $\hat{z}$. This covert signal is then transmitted over the channel after being added to a normal signal x. Similar to every covert communication scheme, there is an observer or warden in the system that will be alerted when seeing any deviation in the statistical properties of the channel. To this end, we are implicitly incorporating a statistical undetectability constraint on the produced covert signals by having a discriminator network used by Willie. The presence of this discriminator network helps us to ensure that these added covert noise signals are distributed as the channel's real noise, making them undetectable.
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 a normal signal, 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 $C(\cdot)$.

1) 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:

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

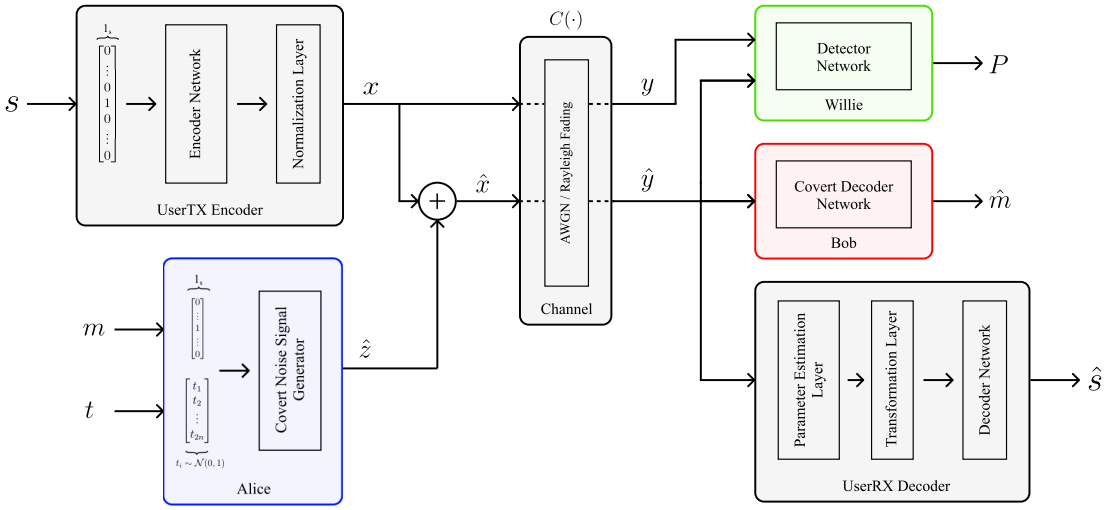


Fig. 1: Our covert communication scheme over AWGN and Rayelgh fading channels on top of a normal autoencoder-based wireless communication system.

2) Rayleigh Fading Channel Output: For the Rayleigh fading channel model, 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:

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

On the receiver side, Bob receives a transmitted signal $\hat{y}$ after the channel noise is added into it. He uses its decoder network to reconstruct the covert message $\hat{m}$. Meanwhile, the UserRX is using the same signal to extract the normal message $\hat{s}$, which is the reconstructed message of s. The statistical properties of signals on the channel are captured by Willie who observes the channel continuously. His objective is to classify sequences of normal y and covert signals $\hat{y}$ that he sees on the channel 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 the same statistical properties.

C. 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 [24]. 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. [25] 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:

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

where $CE(\cdot)$ is the cross entropy between the reconstructed covert message $\hat{m}$ and the actual covert message m. This equation can be used to optimize both for 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 bear in mind that the generated perturbations should have no detrimental impact on the normal communication between UserTX and UserRX, otherwise an unexpected increase in the error rate of communication can be a sign of an abnormal behavior. We apply this constraint by minimizing the autoencoder's loss function during Alice's training:

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

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 parameters are updated.

In our model, the observer entity or Willie, acts as the discriminator in GAN models [13]. The so-called real and fake data in GANs' discriminator training here is mapped to non-covert and covert signals and we define the loss function for

Willie as:

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

where $BCE(\cdot)$ 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.

D. 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 and a more detailed explanation of such a network and the training procedure can be found in the original paper [5]. An overview of the implemented autoencoder network's architecture can be found in table (I). Likewise, table (II) presents details of our covert models.

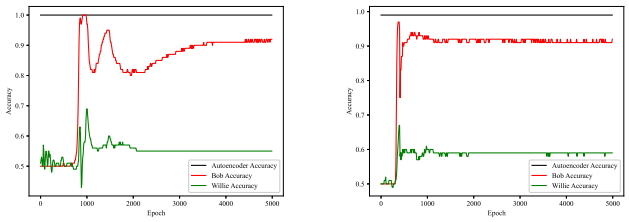
Encoder	
Layer	Output dimension
Input (size 16)	-
Dense + ELU	16
Dense + ELU	2×8
Convolutional (8 filters, kernel size 1×2, stride 1) + Tanh	8×15
Convolutional (8 filters, kernel size 1×4, stride 2) + Tanh	8×6
Convolutional (8 filters, kernel size 1×2, stride 1) + Tanh	8×5
Convolutional (8 filters, kernel 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 I: Autoencoder's network detailed architecture

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 II: Alice, Bob, and Willie's networks detailed architecture

Similar to what is proposed in the original autoencoder wireless communication paper, our autoencoder model accepts a binary message s of size k bits. The encoder part of the model first one-hot encodes the message in order to represent each message as a different class. Then, using its encoder network the message is mapped 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 into. On the receiver side, there will be two different structures given what the channel model is. In case of an AWGN channel model, the signal is simply passed through the decoder network and the intended message gets extracted. For the Rayleigh fading channel model, however, there will be a parameter estimation model that takes in the signal before it passes through the decoder network and estimates the channel's fading coefficients. Followed by this operation, the extracted signal along with the estimated parameters are passed to a transformation function that is supposed to revert the channel effects. In our case, we are using a simple division transformation function that divides the received signal by the estimated channel fading coefficients. Note that more complex transformation functions can be used and are described in [5], 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 by classifying the signal. Similar to the 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. 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. Meanwhile, 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. 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



(a) AWGN channel (b) Rayleigh fading channel

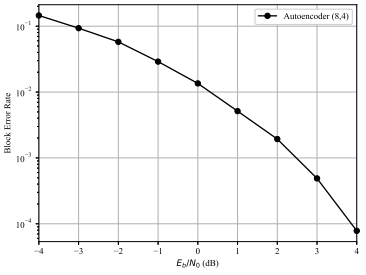
Fig. 2: Covert models' accuracies evaluation during training progress

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 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. 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.

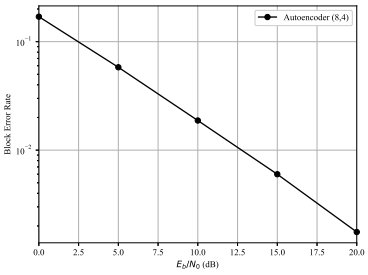
V. Experiments and Evaluation

Before implementing the covert models, we implemented an autoencoder communication network for the normal communication between UserRX and UserTX. Based on the notation used in [5], 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 [5]. 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. The training set contains 8192 random binary messages while this number is 51200 for the test set. We intentionally created a much larger data set for testing to make sure that each symbol y undergoes various channel distortions while evaluating 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.

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 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

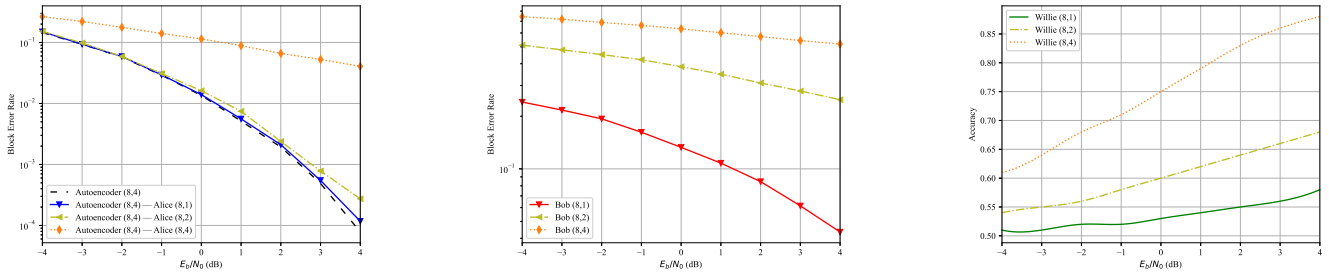


(a) AWGN channel



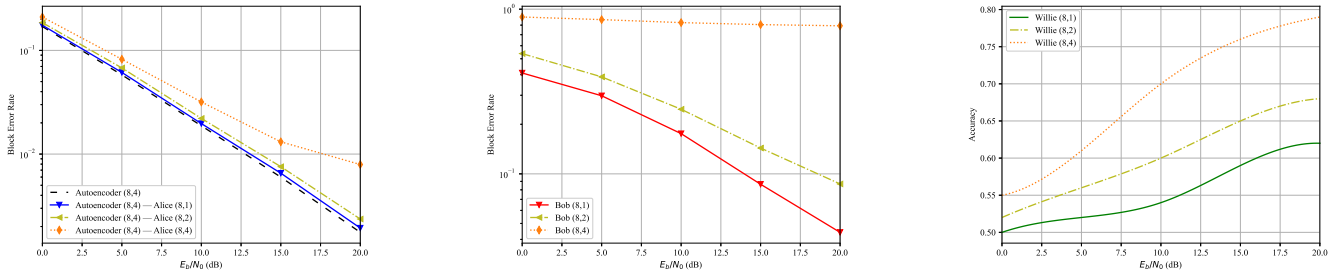
(b) Rayleigh fading channel

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



(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.

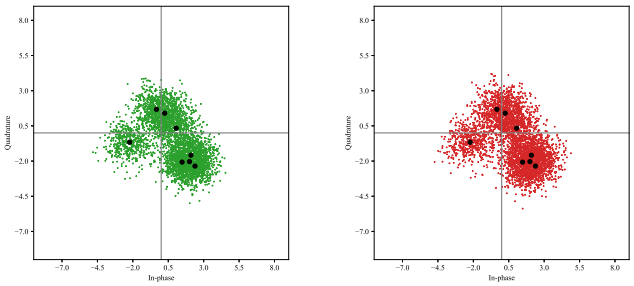


(a) Autoencoder's BLER (b) Bob's BLER (c) Willie's accuracy

Fig. 5: Trained covert models' performance over Rayleigh fading channel for different covert data rates

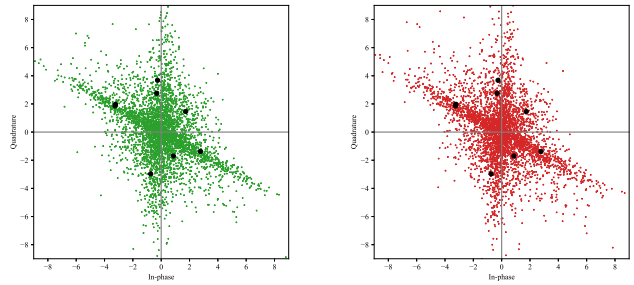
$\lambda_{Willie} = 2\lambda_{Bob} = 4\lambda_{UserRX}$ in (7). We start the training with the learning rate of 0.001 and gradually halve the learning rate after every 500 epochs. In each epoch, we first update parameters of Willie's network using (6), then train Alice's network for one step using (7), 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 the Rayleigh fading channel. Figure (2) 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 communication 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. Figures (4) and (5) show our final results. It also demonstrates how each entity's accuracy of our covert model changes as we increase the covert rate. Expectedly, the covert communication becomes more unreliable, the impact on normal communication increases, and the detection becomes easier by increasing the covert rate. In figures (4(c)) and (5(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 (6) and (7) compare the constellation cloud of a covert and a normal signal for both AWGN and Rayeligh 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 to 15dB for the Rayleigh fading channel so that in both channels, 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



(a) With no covert transmission (b) With covert transmission

Fig. 6: Comparing constellation clouds of a signal before and after our covert scheme being applied on the AWGN channel.



(a) With no covert transmission (b) With covert transmission

Fig. 7: Comparing constellation clouds of a signal before and after our covert scheme being applied on the Rayleigh fading channel.

procedure.

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. We provided information on the performance of covert model on different channel models and noise levels. 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] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

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

[11] 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.

[12] 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.

[13] 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.

[14] 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.

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

[16] 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.

[17] 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.

[18] 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.

[19] 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.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

[24] 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.
[25] 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.
[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.