



ARL-TR-9749 • AUG 2023



Generating Noise-like Communication Codes with Deep Learning

by Michael S Lee

DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited.

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.



Generating Noise-like Communication Codes with Deep Learning

Michael S Lee

DEVCOM Army Research Laboratory

REPORT DOCUMENTATION PAGE

1. REPORT DATE		2. REPORT TYPE		3. DATES COVERED	
August 2023		Technical Report		START DATE 1 October 2019	END DATE 1 March 2022
4. TITLE AND SUBTITLE Generating Noise-like Communication Codes with Deep Learning					
5a. CONTRACT NUMBER		5b. GRANT NUMBER		5c. PROGRAM ELEMENT NUMBER	
5d. PROJECT NUMBER		5e. TASK NUMBER		5f. WORK UNIT NUMBER	
6. AUTHOR(S) Michael S Lee					
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) DEVCOM Army Research Laboratory ATTN: FCDD-RLA-NA Aberdeen Proving Ground, MD 21005				8. PERFORMING ORGANIZATION REPORT NUMBER ARL-TR-9749	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)		11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited.					
13. SUPPLEMENTARY NOTES ORCID ID: Michael S Lee, 0000-0002-0419-6069					
14. ABSTRACT Communication with a low-probability of detection can help prevent targeted jamming and transmitter geolocation. In this report, we use a generative-adversarial network to design symbol communication codes that appear as Gaussian white noise, which is the idealized analogue of expected RF background and thermal circuit noise.					
15. SUBJECT TERMS communication codes; deep learning; generative-adversarial networks; white noise; normal distribution; thermal noise; Network, Cyber, and Computational Sciences					
16. SECURITY CLASSIFICATION OF:				17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 24
a. REPORT UNCLASSIFIED	b. ABSTRACT UNCLASSIFIED	c. THIS PAGE UNCLASSIFIED			
19a. NAME OF RESPONSIBLE PERSON Michael S Lee				19b. PHONE NUMBER (Include area code) (410) 278-5888	

STANDARD FORM 298 (REV. 5/2020)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures	iv
List of Tables	iv
Acknowledgments	v
1. Introduction	1
2. Theory	2
2.1 Ideal Properties of Gaussian White Noise	2
2.2 Discrete Attributes of Randomness	3
2.3 A New Loss Function for a GAN Discriminator	3
3. Methods	4
3.1 GAN	4
3.2 Data	6
3.3 Spectrogram Analysis	7
4. Results	7
5. Discussion and Conclusion	12
6. References	14
List of Symbols, Abbreviations, and Acronyms	16
Distribution List	17

List of Figures

Fig. 1	Quadratic discriminator loss function.....	4
Fig. 2	Schematic of ML system to generate random-like signals.....	5
Fig. 3	Encoder and decoder models. Yellow boxes are convolutional layers. Input text is one-hot encoded prior to first convolutional layer. Legend: <# of kernels> $\times$ (kernel size).....	6
Fig. 4	Discriminator model tries to determine if transmitted message is signal or noise.....	6
Fig. 5	Progress of validation loss and accuracy during training	8
Fig. 6	Histograms of code values for pseudo-random noise ($\mu = 0, \sigma = 12.75$) (top) and ML-generated message noise (bottom)	9
Fig. 7	Comparison of Q–Q plots to test for normality: pseudo-random noise (top) and ML-generated message noise (bottom). The red line indicates an ideal correspondence.....	9
Fig. 8	Autocorrelations for pseudo-random (top) and ML-code (bottom)....	10
Fig. 9	Power spectral densities for pseudo-random noise (top) and ML- generated codes (bottom).....	11
Fig. 10	Magnitude spectrograms for pseudo-random noise (top) and ML- generated codes (bottom).....	11

List of Tables

Table 1	Statistics of signal distributions ($N = 100$).....	12
---------	---	----

Acknowledgments

The author would like to thank Dr John Hyatt, Mr Matthew Ziemann, and Dr Peng Wang for helpful discussions. Supercomputing time was provided by the US Army Combat Capabilities Development Command Army Research Laboratory Supercomputing Resource Center.

1. Introduction

Covert communication over radio involves sending signals that cannot be easily detected or intercepted by third-party observers. Various low-probability of detection (LPD) strategies exist, such as spread spectrum (SS),¹ code-division multiple access (CDMA),² and chaotic CDMA.³ Machine learning (ML) offers a new strategy for achieving LPD.⁴ Specifically, with recent advances in deep learning (DL), including generative-adversarial networks (GANs),² we hypothesize that ML can be used to develop encoding schemes that will be difficult to distinguish from the natural noise—both residing in the RF environment and manifesting in the electronic circuitry of radio receivers.

The earliest documented form of stealth radio communication is SS technology¹ whereby portions of a signal are transmitted over pseudo-randomly selected channels such that an eavesdropper might have little information on any individual band to know what is going on. CDMA¹ is a modern alternative to this strategy that can generate signals below the noise floor (i.e., signal-to-noise ratio [SNR] below 0 dB). Unless the eavesdropper knows the code to look for, as in commercially standard CDMA for mobile phones, they might not be able to recognize the presence of an artificial signal amidst the RF background. Chaotically generated chip sequences³ or long-sequence pseudo-randomly generated keys⁵ can further prevent the eavesdropper from recognizing RF activity. Other methods of LPD communication have been suggested over the years including inexact in-phase/quadrature-phase (I/Q) constellations⁶ and multiple antennas.⁷ The problem of how surreptitious and content-rich the communication can be given the channel conditions has also been analyzed.^{8,9}

GANs are a relatively new development in ML that allow models to synthesize realistic-looking data that resemble the training set.¹⁰ GANs can also be used to generate data that are able to fool a concurrently trained discriminator. In this report, we explore whether it is possible to generate a code that can pass for random Gaussian white noise (GWN) while still accurately conveying English text sequences to a receiver in the know. We train a discriminator neural network model to distinguish between pseudo-random GWN and coded messages. We also simultaneously train encoder and decoder networks to convert between plain-text messages and an 8-bit noise-like code, roughly corresponding to the precision of the I/Q input and output of a typical low-cost software-defined radio.

A fundamental challenge in stealth wireless communication that we are not addressing here is the detectable presence of excess energy in the transmitted frequency band. We do know, however, that SNRs well below 0 dB can be used to

communicate information, as seen in global navigation satellite systems (GNSSs)¹¹ and CDMA applications.

2. Theory

Unlike traditional steganography, where codes are hidden in existing signals,^{12,13} such as over-the-air television, our objective is to hide messages in environmental RFs and electronic circuit noise. For this study, we assume that the composite noise is white (i.e., the same amplitude at all relevant frequencies) and Gaussian (a time-series of normally distributed samples). It should be possible to extend our methods to other noise colors such as pink noise ($1/f$).

2.1 Ideal Properties of Gaussian White Noise

When trying to discern if a signal is ideal GWN, one can look for a few key time-dependent and time-independent properties.¹⁴ For instance, in the limit of long time averaging, the frequency-based power spectrum of white noise should be constant over all relevant frequencies. In addition, the autocorrelation of an ideal GWN time series, $x(t)$, should approach a delta function,

$$\int_0^\tau x(t)x(t_0 - t)dt = \delta(t_0), \quad (1)$$

which indicates that there should be no predictable periodic patterns over all time scales, τ . For time-independent features, the distribution of samples of a GWN signal, $\rho(x)$, should have the form of a Gaussian distribution,

$$\rho(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{\sigma^2}}, \quad (2)$$

where the mean, μ , is usually zero and the standard deviation, σ , is a constant. In this work, σ is set to 0.05, so that sampled values are safely between -1 and 1 . Furthermore, in a Gaussian distribution, the third-order statistic, skew, should be 0, and the fourth-order moment, kurtosis, in the Fisher definition, should also be 0. Moments beyond fourth order can also offer evidence of deviation from ideal GWN. While, in principle, these can be incorporated and validated, we do not consider them in this work. In addition, a quantile–quantile (Q–Q) plot can be used to compare the observed histogram against the expected distribution in each domain segment (quantile).

2.2 Discrete Attributes of Randomness

Another rigorous definition of noise involves assessing the randomness of a set of consecutive samples. While these tests are not incorporated in this work, they would constitute arguably the most stringent evaluation of noise/randomness. A wide variety of randomness tests exist for evaluating sequences of numbers selected from a uniform distribution. Therefore, to test randomness in natural, GWN, one would need to first convert Gaussian noise samples to a uniform distribution. One method would be to remap values with the cumulative density function of the Gaussian, also known as, the standard error function, *erf*:

$$x_{uniform} \leftarrow \Phi(x) = \frac{1}{2} \left[1 + \operatorname{erf} \left(\frac{x-\mu}{\sigma\sqrt{2}} \right) \right]. \quad (3)$$

Among randomness tests to consider would be Diehard, Dieharder, TestU01, and the National Institute of Standards and Technology (NIST) statistical test suite.^{15–18}

2.3 A New Loss Function for a GAN Discriminator

The GAN described in this work is similar in spirit to the proliferation of GAN models in the literature¹⁹ including steganographic GANs.^{12,13} One universal problem with GANs, however, is that their training can have convergence difficulties. In some cases, either the generator or the discriminator dominates. One cause of this problem is that the sigmoidal outputs of the discriminators become trapped in local minima. In this work, we give the discriminator output an unbounded quadratic loss function,²⁰

$$L_{critic} = -2 \left(y_{true} - \frac{1}{2} \right) \left(y_{pred} - \frac{1}{2} \right) + \left(y_{pred} - \frac{1}{2} \right)^2, \quad (4)$$

where a value of 0 implies the data is real (i.e., similar to natural Gaussian noise), and 1 means the data is fake (i.e., similar to artificial codes). As can be seen in Fig. 1, the loss function is quadratic in both cases of false and real ground truth messages. The total adversarial loss function is a sum of this term and the standard categorical cross-entropy loss of the message reconstruction, L_{recon} .

$$L_{adversarial} = L_{recon} + cL_{discriminator}, \quad (5)$$

and c is a constant that can be adjusted by the user to equalize the reconstruction and discriminator losses and ensure the input and output messages are nearly identical. In this work, c is set to a value of 4.

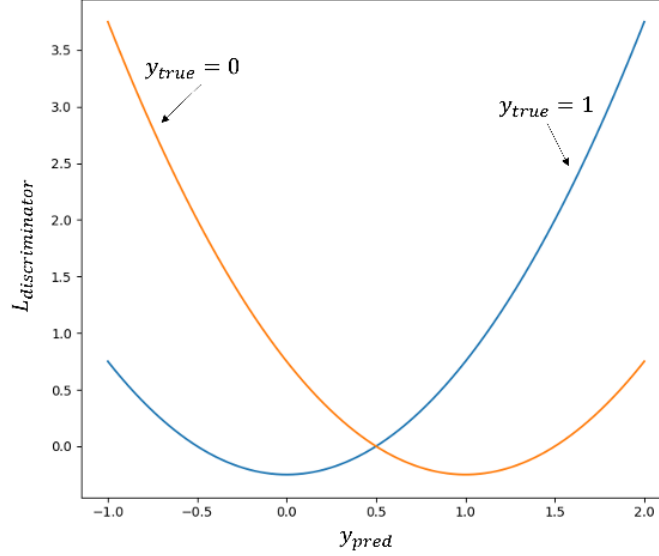


Fig. 1 Quadratic discriminator loss function

3. Methods

3.1 GAN

Figure 2 schematically illustrates the automated code design and analysis method used in this work. English text messages with 16 characters apiece are converted to one-hot matrices of all possible bytes ($shape = [16, 256]$) and then are deterministically encoded by an encoder and decoded by a decoder back into one-hot matrices. The intermediate synthetic code is assessed by a discriminator that decides whether it is an encoded message or noise. Because the autoencoder learns how to encode/decode 16 symbols at-a-time, the model acquires some notions of the probability densities of the English language. In the extreme, a customized ML model could be optimized specifically on the common messages expected to be used for that system.

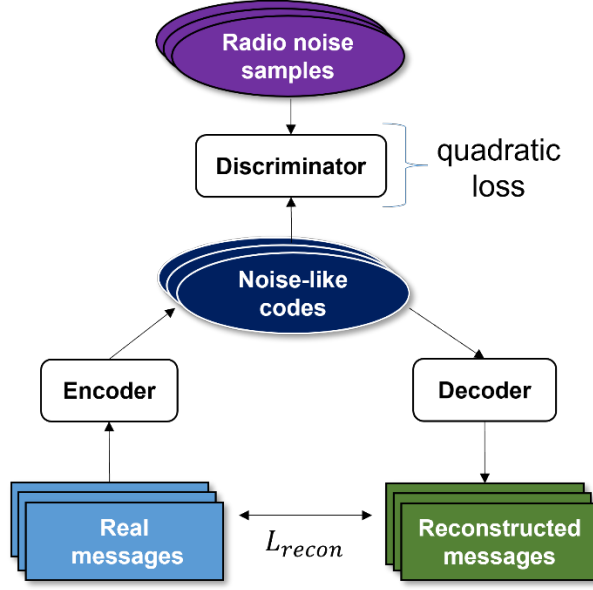


Fig. 2 Schematic of ML system to generate random-like signals

Samples of pseudo-random (i.e., noise) vectors and encoded messages are used to train the discriminator. The quadratic discriminator specified in Section 2.3 is the discriminator’s only loss function. The generator model is a composite of the encoder–decoder (a.k.a., autoencoder) model with a discriminator model. The topologies for each model are detailed in Figs. 3 and 4. The soft quantize layer, $Q(x)$, in Fig. 3, is a custom layer that converts the normalized and scaled input into 256 quantiles:

$$Q(x) = x - \frac{1}{128\pi} \tan^{-1}[\tan(128\pi x)]. \quad (6)$$

The input x is pre-scaled with a factor of 0.05 to almost guarantee it will be within an interval of $[-1,1]$.

The discriminator and autoencoder models are trained iteratively: one cycle of discriminator training, followed by one cycle of the autoencoder. The adversarial model is trained having fixed weights for the discriminator and the samples all being labeled as “true random.” This encourages the autoencoder to create synthetic codes that can fool the discriminator into believing the codes are true noise.

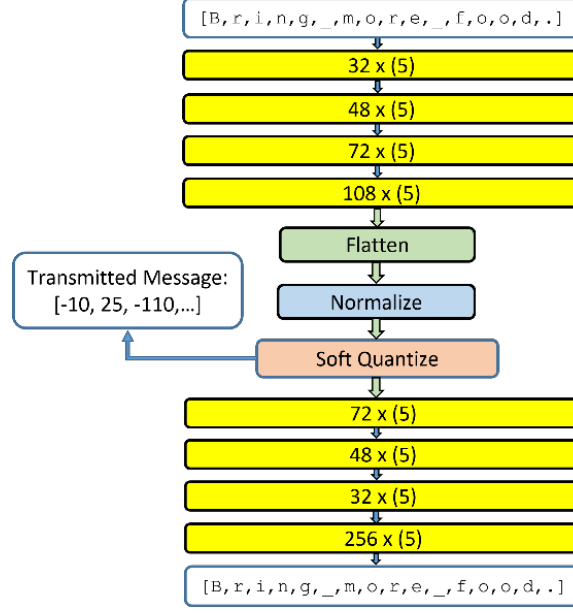


Fig. 3 Encoder and decoder models. Yellow boxes are convolutional layers. Input text is one-hot encoded prior to first convolutional layer. Legend: <# of kernels> × (kernel size).

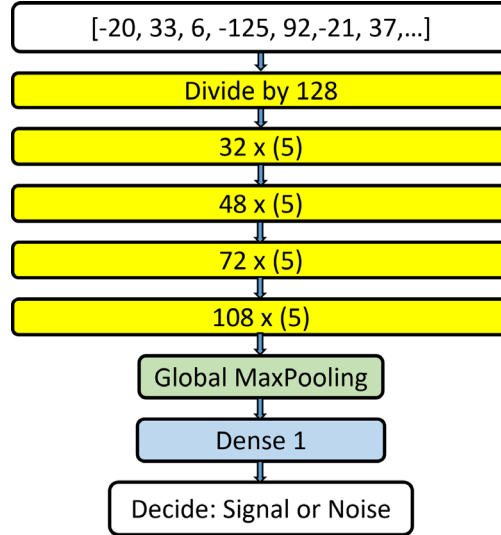


Fig. 4 Discriminator model tries to determine if transmitted message is signal or noise

3.2 Data

For text messages in this work, we used the popular 20 Newsgroups data set,²¹ which is a collection of 18,000 English-language bulletin board posts from 20 diverse topics. A large set of the posts with headers removed were converted to 27,000 individual sentences. We designated the first 16 characters of each sentence as single messages. For training, 80% of the data was randomly allocated, and for validation, 20% was randomly allocated.

3.3 Spectrogram Analysis

For analysis, spectrograms were generated with the short-time Fourier transform using the Python function, `scipy.signal.stft`,²² with a window size of 128 samples, and overlap of 64 samples. Magnitude spectrograms are the complex absolute value of each point in time and frequency.

4. Results

In Fig. 5, we show the progress of validation loss and accuracy as a function of training epoch, where one epoch equals 100 batches of size 100. We can see that transmission validation accuracy asymptotes to approximately 98%. The noise discriminator’s accuracy remains above 50% throughout, leveling off around 60%. The steadiness of the noise discriminator during training implies that it remained an effective constraint while the autoencoder was optimizing. Also, we can see that the discriminator loss fluctuated around a value of 0, which is consistent with the quadratic loss graph in Fig. 1. In addition to the noise-free channel assumption, we trained models (results not shown) in both 0.5× and 1.0× noise channels to see how they would perform in “the real world” and found that message accuracy went down to 94% at 0.5× noise and plummeted to 70% error at 1.0× noise (equivalent to an SNR of 0 dB).

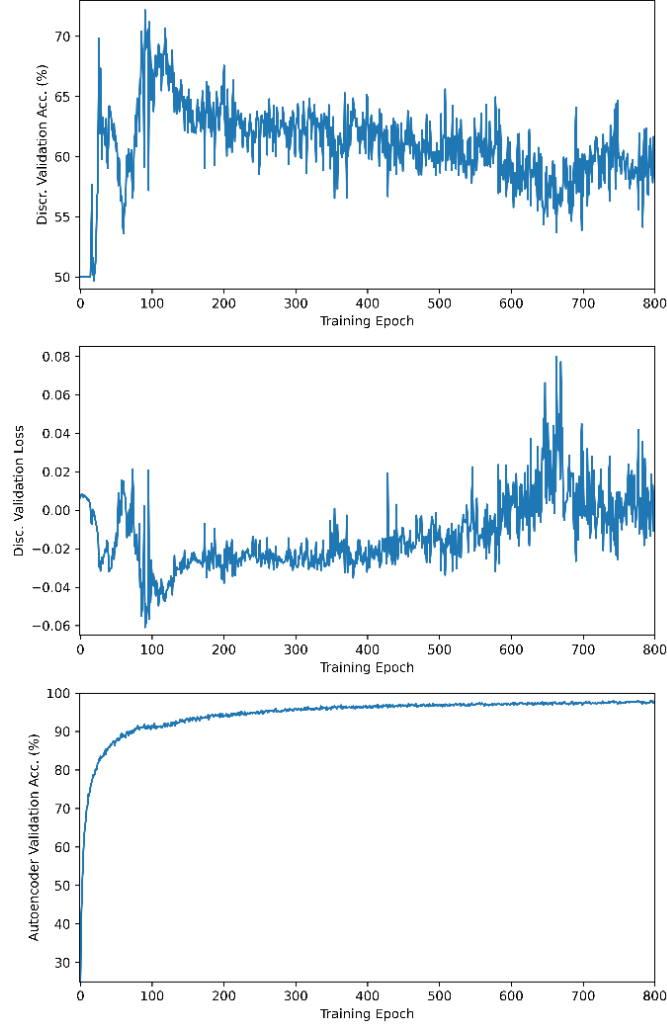


Fig. 5 Progress of validation loss and accuracy during training

Next, we use a variety of metrics to compare pseudo-random white noise to the codes used to “transmit” the text messages. In Fig. 6, histograms of pseudo-random noise and codes both resemble a Gaussian distribution. The computed symbol bit entropy for both sets is 4.73. More precisely, in Fig. 7, we present Q–Q plots, which further validate that the distribution of ML-generated codes closely resembles a normal distribution. Only the tails are slightly off, but not much worse than a pseudo-random generated distribution using Python, `numpy.randn`.²³

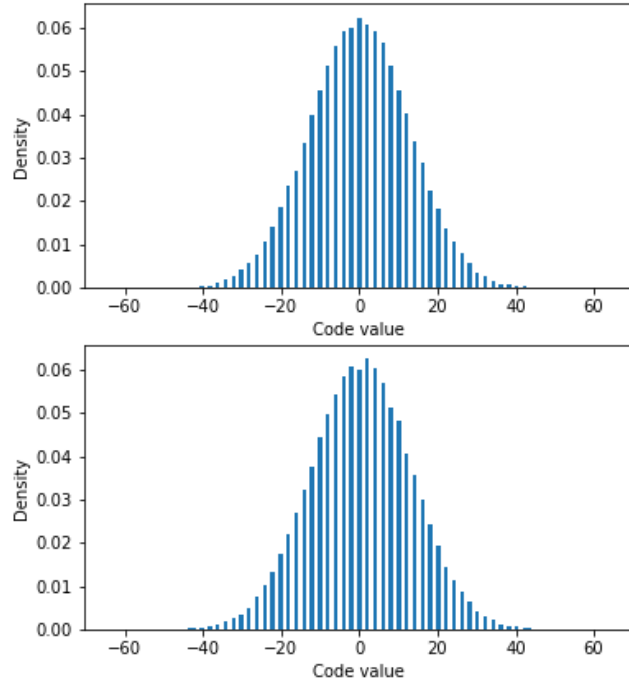


Fig. 6 Histograms of code values for pseudo-random noise ($\mu = 0, \sigma = 12.75$) (top) and ML-generated message noise (bottom)

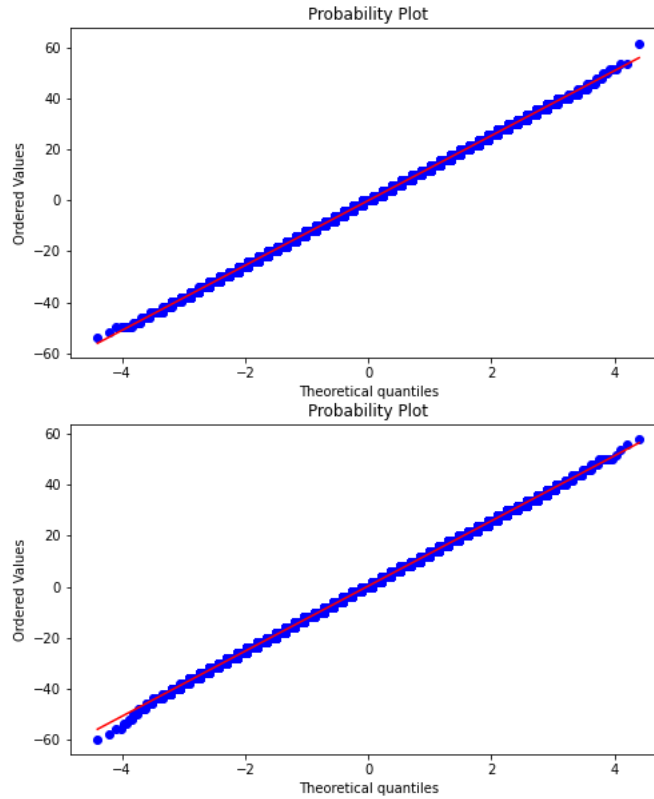


Fig. 7 Comparison of Q-Q plots to test for normality: pseudo-random noise (top) and ML-generated message noise (bottom). The red line indicates an ideal correspondence.

In Fig. 8, we look for any periodicities in the encoded signal using autocorrelation. Both signals appear to have no significant time-delayed correlates. In Fig. 9, we observe that both the pseudo-random noise and the ML-codes have a flat frequency distribution, with the message noise slightly bumpier. Figure 10 shows the time-frequency (spectrogram) representations of the noise and the ML-generated code for multiple messages. What is apparent for the codes is not so much patterns but stronger “shadows” indicating larger inter-sample variations than the regular noise, which is not apparent in the raw time sample histogram (Fig. 6).

Finally, in Table 1 we evaluate statistics of the ML-generated codes for encoding 100 “messages.” This analysis provides further evidence that the message codes have Gaussian noise-like statistics.

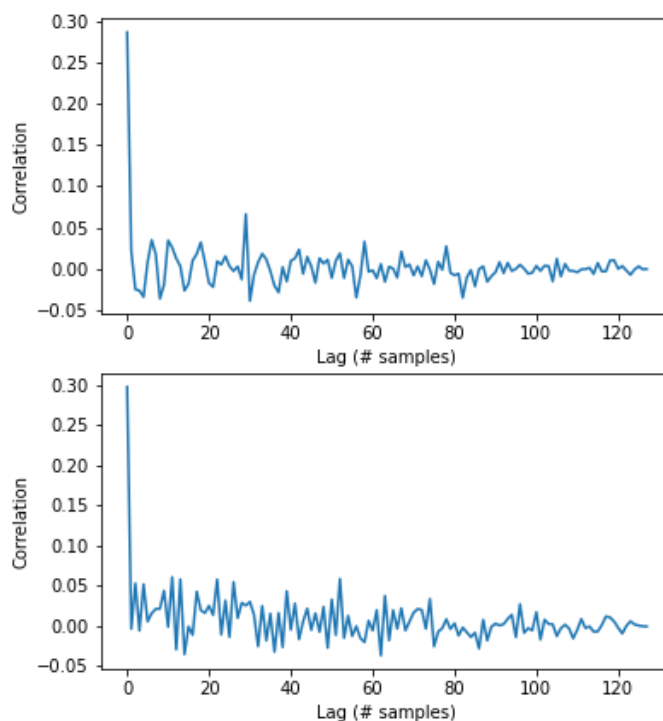


Fig. 8 Autocorrelations for pseudo-random (top) and ML-code (bottom)

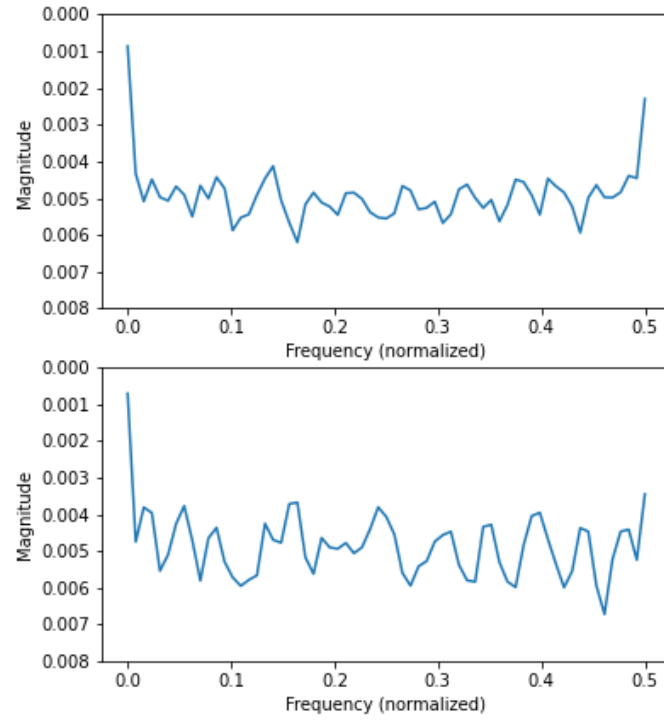


Fig. 9 Power spectral densities for pseudo-random noise (top) and ML-generated codes (bottom)

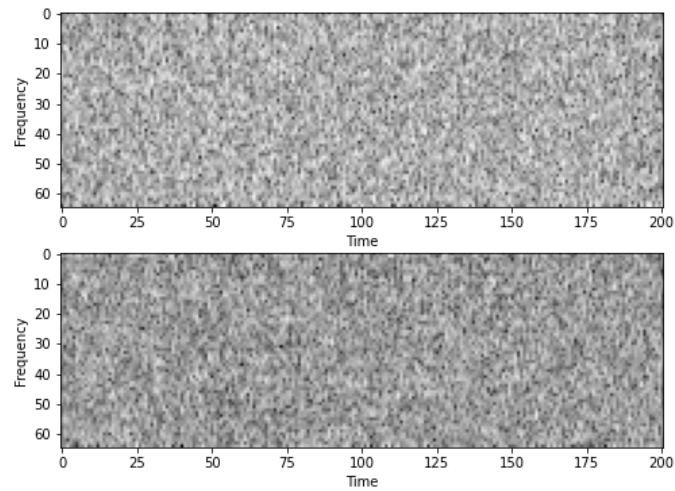


Fig. 10 Magnitude spectrograms for pseudo-random noise (top) and ML-generated codes (bottom)

Table 1 Statistics of signal distributions ($N = 100$)

Metric	Pseudo-random	ML-code
mean	-0.00002 (0.00429)	0.00068 (0.00414)
standard deviation	0.0494 (0.0031)	0.0498 (0.0034)
skew	-0.034 (0.234)	0.005 (0.226)
kurtosis (Fisher)	0.023 (0.434)	-0.023 (0.434)
autocorrelation (largest peak)	0.17 (0.03)	0.18 (0.04)

5. Discussion and Conclusion

This work builds on the scientific literature for covert communication³ and ML-based steganography.^{12,13} Unlike most previous covert communication research, the computer, not a human, developed a modulation code to fool a concurrently trained artificial signal detection system. Unlike traditional steganography, the code is designed to hide in the environmental and circuit noise, rather than in an ordered and/or expected artificial signal such as a television broadcast.

Traditional SS methods are not so much hiding the signal through modulation as much as they are distributing rare signals in time over a wide range of frequencies. With the advent of software-defined radios, it might be possible to identify SS signals by long-term visualization or analysis of broad-spectrum waterfalls. On the other hand, CDMA methods are more likely to be discreet, given how weak their SNR can be. To avoid detection, long and unpredictable duration codes with a private “key,” such as what might be used by military GNSS,¹¹ may be hard to find in the noise.

While we have demonstrated that a deterministic autoencoder can both transmit and receive reliable messages in a mildly noisy channel while appearing to have Gaussian noise statistics, the following question remains: *Can we guarantee fooling an adversary’s discriminator ML model?* This will depend on, in part, how often the communication code is changed, and whether a particular GAN model leaves a distinctive signature in its generated codes each time it is trained. The advantage, we conjecture, will always remain on the side of the communicators who knows what code to look for and how to decode it in the seeming randomness of the background RF spectrum. The adversary must at some point limit its hypotheses in guessing which emitter directions and locations should be investigated and mathematically processed further.

Suggestions for future work include determining whether ML models produce a stealth of similar quality as private-key or chaotic CDMA. In addition, ML encoding could be implemented on a real radio to determine RF effects not considered here, such as the inherent periodicity of digital impulses. Finally, we

conjecture that repeating the same message too many times may lead to signal periodicity; thus, obfuscation techniques such as time-varying message encryption may be necessary to ensure LPD.

6. References

1. Torrieri D. Principles of spread-spectrum communication systems. 4th ed. 2018.
2. Gold R. Optimal binary sequences for spread spectrum multiplexing (Corresp.). IEEE Transactions on information theory. 1967;13(4):619–621.
3. Quyen NX, Nguyen CT, Barlet-Ros P, Dojen R. A novel approach to security enhancement of chaotic DSSS systems. 2016 IEEE Sixth International Conference on Communications and Electronics (ICCE), IEEE; 2016, 471–476.
4. Qasem ZA, Esmail H, Sun H, Qi J, Wang J. Deep learning-based spread-spectrum FGSM for underwater communication. Sensors. 2020;20(21):6134.
5. Sagey WE. Featureless covert communication system. <https://patents.google.com/patent/US6252962B1/en>.
6. Dutta A, Saha D, Grunwald D, Sicker D. Secret agent radio: covert communication through dirty constellations. International Workshop on Information Hiding, Springer; 2012.
7. Zhang W, Zhao N, Zhang S, Yu FR. Multi-antenna covert communications with random access protocol. arXiv preprint arXiv:1907.07481; 2019.
8. Bloch MR. Covert communication over noisy channels: a resolvability perspective. IEEE Transactions on Information Theory. 2016;62(5):2334–2354.
9. Hill PCJ, Adams ER, Comley VE. Techniques for detecting and characterising covert communication signals. 1997;57–60.
10. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial nets. Advances in Neural Information Processing Systems; 2014.
11. Hofmann-Wellenhof B, Lichtenegger H, Wasle E. GNSS—global navigation satellite systems: GPS, GLONASS, Galileo, and more, Springer Science & Business Media; 2007.
12. Chu C, Zhmoginov A, Sandler M. CycleGAN, a master of steganography. arXiv preprint arXiv:1712.02950; 2017.
13. Hayes J, Danezis G. Generating steganographic images via adversarial training. Advances in Neural Information Processing Systems; 2017.

14. Uchino M, Tagiri O, Shinozuka T. Real-time measurement of noise statistics. *IEEE Transactions on Electromagnetic Compatibility*. 2001;43(4):629–636.
15. Bassham III LE, Ruhkin AL, Soto J, Nechvatal JR, Smid ME, Leigh SD, Levenson M, Vangel M, Heckert NA, Banks DL. A statistical test suite for random and pseudorandom number generators for cryptographic applications. National Institute of Standards and Technology (US); 2010 Sep 16. Report No.: NIST SP - 800-22 Rev 1a.
16. Soto J. Statistical testing of random number generators. *Proceedings of the 22nd National Information Systems Security Conference*. 1999;10(99).
17. Brown RG, Eddelbuettel D, Bauer D. Dieharder: a random number test suite. Open Source software library, under development. <http://www.phy.duke.edu/~rgb/General/dieharder.php>, 2013.
18. L’Ecuyer P, Simard R. TestU01: AC library for empirical testing of random number generators. *ACM Transactions on Mathematical Software (TOMS)*. 2007;33(4):1–40.
19. Gui J, Sun Z, Wen Y, Tao D, Ye J. A review on generative adversarial networks: Algorithms, theory, and applications. *arXiv preprint arXiv:2001.06937*; 2020.
20. Lee MS. Deep learning restoration of signals with additive and convolution noise. *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications III*, vol. 11746, International Society for Optics and Photonics; 2021, 1174617.
21. Lang K. Newsweeder: learning to filter netnews. *Proceedings of the 12th International Conference on Machine Learning*; 1995, 331–339.
22. Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, Burovski E, Peterson P, Weckesser W, Bright J, et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*. 2020;17(3):261–272.
23. Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, et al. Array programming with NumPy. *Nature*. 2020;585:357–362.

List of Symbols, Abbreviations, and Acronyms

CDMA	code division multiple access
DL	deep learning
GAN	generative-adversarial network
GNSS	global navigation satellite system
GWN	Gaussian white noise
I/Q	in-phase/quadrature-phase
LPD	low-probability of detection
ML	machine learning
NIST	National Institute of Standards and Technology
Q-Q	quantile-quantile
RF	radio frequency
SNR	signal-to-noise ratio
SS	spread spectrum

1 DEFENSE TECHNICAL
(PDF) INFORMATION CTR
DTIC OCA

1 DEVCOM ARL
(PDF) FCDD RLB CI
TECH LIB

12 DEVCOM ARL
(PDF) FCDD RLA
A SWAMI
FCDD RLA N
M FRAME
B RIVERA
FCDD RLA NA
M LEE
E MARK
M ZIEMANN
B KRACZEK
FCDD RLA NB
R DROST
FCDD RLA NC
M VINDIOLA
P WANG
FCDD RLA ID
J JOHNSON
FCDD RLR EI
J HYATT