JOINT CODING AND MODULATION IN THE ULTRA-SHORT BLOCKLENGTH REGIME FOR BERNOULLI-GAUSSIAN IMPULSIVE NOISE CHANNELS USING AUTOENCODERS

Kirty Vedula, Randy Paffenroth, and D. Richard Brown III

Worcester Polytechnic Institute, 100 Institute Road, Worcester, MA 01609

ABSTRACT

This paper develops a joint coding and modulation scheme for end-to-end communication system design using an autoencoder architecture in the ultra-short blocklength regime. Unlike the classical approach of separately designing error correction codes and modulation schemes for a given channel, the approach here is to learn an optimal mapping directly from messages to channel inputs while simultaneously learning an optimal mapping directly from channel outputs to estimated messages. The block error rate (BLER) of this approach is compared against classical short blocklength linear block codes with binary phase shift keying (BPSK) modulation in additive white Gaussian noise (AWGN) and Bernoulli-Gaussian impulsive noise (BGIN) channels. For AWGN channels, numerical results show that the autoencoder can achieve better BLER performance than BPSK modulated Hamming codes with maximum likelihood decoding. For BGIN channels, numerical results show the autoencoder achieves uniformly better BLER performance than conventional block codes with BPSK modulation, even with impulsive noise mitigation techniques such as blanking and clipping. The proposed architecture is general and can be modified for comparison against other block coding schemes and higher-order modulations.

Index Terms— Joint coding and modulation, impulsive noise channel, error control coding, autoencoder, short blocklength, IoT

1. INTRODUCTION

Communication systems have traditionally been designed by considering each block, i.e., channel encoding, modulation, demodulation, and channel decoding, separately [1]. For example, coding theorists typically design error correction codes by abstracting and lumping the effects of modulation, physical channel impairments, and demodulation into an "effective channel" with certain statistical properties. Similarly, communication theorists typically ignore error correction coding and develop efficient and robust modulation schemes to overcome physical channel impairments. These approaches are suboptimal, however, in many cases. This paper considers the question of what can be gained by jointly designing error correction coding and modulation schemes with a focus on the ultra-short blocklength regime with a small number of input bits per message, i.e., (n, k) block codes with $k \leq 16$. This regime is also of contemporary interest due to Internet of Things (IoT) devices often transmitting only infrequent and short messages with low latency requirements.

Joint channel coding and modulation was first proposed in the context of trellis coded modulation and shaping [2] and more recently in probabilistic amplitude shaping [3] for low-density parity-check codes. More recently, researchers have considered the use of *autoencoders* for jointly designing coding and modulation schemes to overcome channel impairments [4, 5]. Generally speaking, autoencoders can be used to find a low-dimensional representation of the input while facilitating reconstruction at the output with minimal error [6]. Autoencoders have been successfully applied for end-to-end communication system design in the binary-input additive white Gaussian noise (bi-AWGN) channel [7–10]. More recently, autoencoders have been used to automate the discovery of decoding algorithms for channels that do not have known good codes, e.g., the feedback channel [11]. This work demonstrates the strong generalization capability of classical algorithms like Viterbi and BCJR on convolutional and turbo codes, with near-optimal performance on AWGN channels. The adaptability and flexibility of neural networks allow them to operate in situations where some simplifying assumptions of standard coding, modulation, demodulation and decoding techniques are not fulfilled [12]. Specifically, autoencoders can be play a useful role in settings where there is either a model deficit (when the model is not well-understood) or algorithmic deficit (when the model is understood, but solutions are difficult to find in a large search space). This paper focuses on the latter setting.

While the focus of this paper is on Bernoulli-Gaussian impulsive noise (BGIN) channels [13], we first demonstrate the efficacy of an autoencoder on AWGN channels. Numerical results show that, given the same block code parameters (n, k) and the same number of channel symbols, the autoencoder can learn a code that achieves better BLER performance than Hamming codes with BPSK modulation and maximum likelihood (soft decision) detection. Somewhat unexpectedly, the autoencoder finds a code with a smaller minimum Euclidean distance but similar or better overall BLER than BPSK modulated Hamming codes.

The main contribution of this paper is the application of the ideas in [4, 5, 7–10] toward the development of new codes for the BGIN channel which, to the best of our knowledge, has not been studied in this context. Impulsive noise is prevalent in interference from machines and/or electronic devices with random and high power noise [14, 15] and little is currently known about channel coding in this setting [16, 17]. In fact, soft decision decoding, while optimal in the AWGN channel, can perform worse than hard decision decoding in impulsive noise channels. This has led to the development of impulsive noise mitigation techniques such as blanking and clipping [18–20]. Rather than employing these heuristic techniques, our approach is to train an autoencoder in impulsive noise channels to minimize the BLER. Numerical results show that the trained autoencoder uniformly outperforms classical block codes with BPSK modulation in the BGIN channel even when impulsive noise mitigation techniques such as blanking and clipping are employed. The proposed architecture is general and can be modified for comparison against other block coding schemes and higher-order modulations.

2. SYSTEM MODEL

We assume the point-to-point communication system model with $M = 2^k$ distinct messages as illustrated in Fig. 1. The classical approach at the transmitter is to provide a block of k bits at the input of the channel encoder, map these bits to an n-bit codeword, and then map this codeword to m real-valued symbols for transmission through the channel. Similarly, at the receiver, the noisy symbols are first demodulated and channel decoding is performed either on the soft demodulator outputs or on the hard decisions from the demodulator. The autoencoder considered here lumps the coding and modulation functions into $f_\theta : \{0,1\}^k \mapsto \mathbb{R}^m$, the memoryless channel function into $g : \mathbb{R}^m \mapsto \mathbb{R}^m$, and the demodulation and decoding functions into $h_\theta : \mathbb{R}^m \mapsto \{0,1\}^k$. The subscript θ indicates that these functions have parameters that we can adapt and learn to achieve a certain goal, e.g., minimizing the BLER.

We design the encoder and decoder similar to [4, 5] as shown in Figure 1. The transmitter seeks to communicate message $q \in \{1,\ldots,M\}$ to the receiver. The message is first mapped using a one-hot-encoding scheme to $\boldsymbol{s} = \mathbf{1}_q$ where $\mathbf{1}_q \in \mathbb{R}^M$ is a standard basis vector with q^{th} element equal to one and all other elements equal to zero. Let $\mathbb{Q} = \cup_{q=1}^{M} \mathbf{1}_q \subset \mathbb{R}^M$. The transmitter neural network then generates a channel input $\boldsymbol{s}_e = f_\theta(\boldsymbol{s})$ where $f_\theta : \mathbb{Q} \mapsto \mathbb{R}^m$ and where θ represents the weight vectors and biases. The channel input $\boldsymbol{s}_e$ is then sent through a mapping $\boldsymbol{y} = g(\boldsymbol{s}_e)$ with $g : \mathbb{R}^m \mapsto \mathbb{R}^m$ where a per-block energy constraint is imposed and an impairment (typically noise) is applied. The receiver then applies the transformation $h_\theta : \mathbb{R}^m \mapsto \mathbb{R}^M$ to compute a posterior probability vector $\boldsymbol{p} \in \mathbb{R}^M$ of all possible messages $q \in \{1,\ldots,M\}$ given $\boldsymbol{y}$. The decoded message $\hat{q}$ is simply the index of the maximum element of $\boldsymbol{p}$. The autoencoder is trained end-to-end to minimize the categorical cross-entropy loss function $\mathcal{L}$ between $\boldsymbol{s}$ and $\boldsymbol{p}$ with respect to θ.

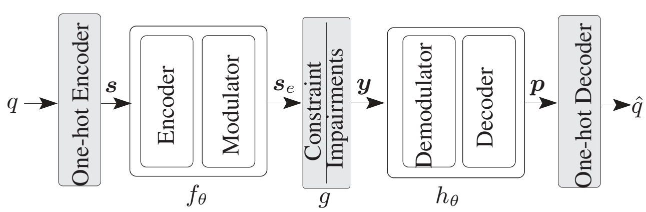

Fig. 1. An autoencoder for an end-to-end communication system.

The focus in this paper is on comparisons to BPSK-modulated Hamming codes where $(n,k) = (2^\ell - 1, 2^\ell - 1 - \ell)$ for $\ell = 3,4,\ldots$ and $m = n$. For fair comparisons, the autoencoder uses parameters (n,k,m) identical to those of the conventional coding and modulation scheme and is subject to the same per-block total energy constraint as conventional coding and modulation.

2.1. Block AWGN Channel

The AWGN channel is a mapping $Y = g(X) = X + Z$ where $X \in \mathcal{X}^m$ and $Z \sim \mathcal{N}(0, \sigma^2 I_m)$. For BPSK modulated symbols, $\mathcal{X} = \{-\sqrt{\mathcal{E}_c}, +\sqrt{\mathcal{E}_c}\}$ where $\mathcal{E}_c = k\mathcal{E}_b/n$ is the energy per bit of the modulated coded signal and $\mathcal{E}_b$ is the energy per information bit. Such a channel can be compactly represented as $\text{AWGN}(\mathcal{E}_b/N_0)$ where $\mathcal{E}_b/N_0$ is the SNR of the AWGN channel.

2.2. Block BGIN Channel

The BGIN channel can be represented by $Y = g(X) = X + (I_m - B)Z_0 + BZ_1$ where $X \in \mathcal{X}^m$, $Z_0 \sim \mathcal{N}(0, \sigma_0^2 I_m)$, $Z_1 \sim \mathcal{N}(0, \sigma_1^2 I_m)$ and $B = \text{diag}(b_1,\ldots,b_m)$ where b_i are i.i.d. Bernoulli random variables with $b_i = 1$ with probability p_b and $b_i = 0$ otherwise. In typical impulsive noise channels we assume $\sigma_1^2 \gg \sigma_0^2$ such that p_b represents the probability of the occurrence of high variance impulsive noise for a given channel input. Such a channel can be compactly represented as $\text{BGIN}(\mathcal{E}_b/N_0, \mathcal{E}_b/N_1, p_b)$ where $\mathcal{E}_b/N_0$ is the SNR when the noise has low variance, $\mathcal{E}_b/N_1$ is the SNR when the noise has high variance, and p_b is the probability the high noise variance channel.

3. AUTOENCODER FOR AWGN CHANNELS

To illustrate the utility of an autoencoder in a simple setting, we first consider joint modulation and encoding with respect to a BPSK-modulated Hamming code in AWGN channels. The network is trained using the Adam optimizer [21] with a categorical cross-entropy loss function at a training SNR of $\mathcal{E}_b/N_0 = 3$ dB. The schematic of the autoencoder is shown in Table 1.

Table 1. Schematic of the autoencoder.

Layer	Output Dim.	Parameters
input	M	0
fully connected + ReLU	M	272
fully connected + linear	n	119
Energy Constraint	n	0
AWGN/Impulsive Channel	n	0
fully connected + ReLU	M	128
fully connected + softmax	M	272

We first trained the autoencoder for the case $(n,k,m) = (7,4,7)$ with 10^6 examples, 2000 epochs, training SNR 3 dB and a batch size of 1000. Table 2 compares the learned autoencoder codebook (columns 8-14) to a $(7,4)$ Hamming code with BPSK modulation [22] (columns 1-7), both with a total energy per encoded block set to $\mathcal{E} = 7$. Note that the learned codewords have the same total per-block energy as conventional BPSK-modulated $(7,4)$ Hamming codes, but the elements of each learned codeword are not constant modulus.

Table 2. BPSK-modulated Hamming $(7,4)$ codewords and the learned $(7,4,7)$ autoencoder codewords, both with $\mathcal{E} = 7$.

-1	-1	-1	-1	-1	-1	-1	0.60	0.37	1.30	-0.68	-1.15	0.40	1.69
-1	-1	-1	1	-1	1	1	0.89	-0.32	-0.06	-1.66	-1.57	-0.82	-0.43
-1	-1	1	-1	1	1	1	-1.40	-0.58	-0.88	0.58	0.65	-0.54	-1.70
-1	-1	1	1	1	-1	-1	0.71	-0.93	-1.92	0.08	0.13	1.22	-0.64
-1	1	-1	-1	1	1	-1	-1.50	-0.44	1.44	-1.41	0.14	-0.61	-0.29
-1	1	-1	1	1	-1	1	-0.17	-0.99	0.01	-0.29	1.90	-1.16	0.98
-1	1	1	-1	-1	-1	1	-1.59	0.21	-0.99	-0.47	-0.68	-1.48	0.76
-1	1	1	1	-1	1	-1	0.29	0.68	1.71	0.46	0.19	-1.00	-1.51
1	-1	-1	-1	1	-1	1	-0.66	0.40	-1.19	1.68	0.68	0.37	1.25
1	-1	-1	1	1	1	-1	0.10	1.41	-1.46	-0.67	-1.15	0.92	0.48
1	-1	1	-1	-1	1	-1	1.79	-0.60	0.92	1.10	0.72	0.37	0.84
1	-1	1	1	-1	-1	1	0.88	1.83	-0.05	0.33	0.57	-1.29	0.88
1	1	-1	-1	-1	1	1	0.68	-0.78	1.08	0.04	-0.91	1.39	-1.42
1	1	-1	1	-1	-1	-1	-0.64	-1.97	-0.54	-0.30	-0.81	0.68	1.10
1	1	1	-1	1	-1	-1	-0.97	-0.79	0.72	0.28	1.27	1.79	0.11
1	1	1	1	1	1	1	-1.79	0.23	0.41	0.87	-1.53	0.69	-0.03

Table 3 shows the pairwise Euclidean distance statistics of conventional Hamming BPSK-modulated codewords and of the autoencoder learned codewords in Table 2. Observe that, although the codewords learned by the autoencoder do not exhibit the structure of the BPSK-modulated Hamming $(7,4)$ code, the distance statistics of the autoencoder are essentially identical to those of the Hamming code. Also note that the learned codewords are not unique, i.e., retraining will result in different learned codewords. Nevertheless, in each test, the distance statistics after training were always consistent with those in Table 3.

Table 3. Pairwise Euclidean distance statistics for BPSK-modulated Hamming (n, k) and (n, k, n) autoencoders with $\mathcal{E} = n$.

Scheme	Min	Mean	Max
Hamming $(7,4)$ with BPSK	3.464	3.836	5.292
$(7,4,7)$ Autoencoder	3.429	3.836	5.289
Hamming $(15,11)$ with BPSK	3.464	5.430	7.746
$(15,11,15)$ Autoencoder	3.277	5.431	7.609

Figure 2 plots the achieved BLER of the $(7,4,7)$ autoencoder in AWGN channels. The BLER of BPSK-modulated Hamming $(7,4)$ codes and several finite-blocklength bounds (theoretical RCU [23] and metaconverse [24]) along with the normal approximation [25] are also plotted for comparison. In this case, the BLER performance of the autoencoder is essentially identical to a BPSK-modulated Hamming $(7,4)$ code with soft decision (maximum likelihood) decoding.

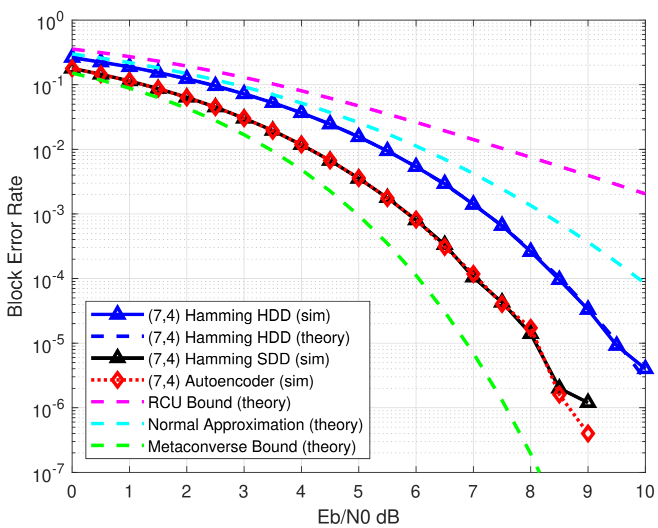

Fig. 2. BLER of the trained $(7,4,7)$ autoencoder and BPSK-modulated Hamming $(7,4)$ in an AWGN(E_b/N_0) channel.

These tests were repeated for a $(15,11,15)$ autoencoder and compared to BPSK-modulated Hamming $(15,11)$ codes. The learned codebook is too large to present here, but the distance statistics are shown in Table 3 with a total per-block energy constraint of $\mathcal{E} = 15$. In our $(15,11,15)$ autoencoder tests, we observed that the autoencoder quickly converged to codewords with mean distance identical to BPSK-modulated $(15,11)$ Hamming codes, whereas the minimum distance was much slower to converge. The results shown in Table 3 were achieved after training on 2×10^7 examples for 150 epochs (other training parameters same as previous example).

Figure 3 plots the achieved BLER of the $(15,11,15)$ autoencoder in AWGN channels along with the BLER of BPSK-modulated Hamming $(15,11)$ code and the finite-blocklength bounds and approximations. Somewhat surprisingly, in light of the autoencoder's worse minimum distance statistic in Table 3, the achieved BLER of the $(15,11,15)$ autoencoder is approximately 0.5 dB better than that of the conventional BPSK-modulated Hamming $(15,11)$ code with soft decision (maximum likelihood) decoding. Additional inspection of the *conditional* BLER for each learned codeword showed that the autoencoder learned an *asymmetric* code in the sense that certain codewords had worse conditional BLER and other codewords had better conditional BLER than the unconditional BLER. This is in contrast to Hamming codes with BPSK modulation where each codeword has a conditional BLER matching the unconditional BLER due to symmetry. By finding more codewords with good conditional BLER (at the cost of a small number of codewords with worse conditional BLER), the autoencoder can outperform Hamming codes with BPSK modulation in terms of unconditional BLER.

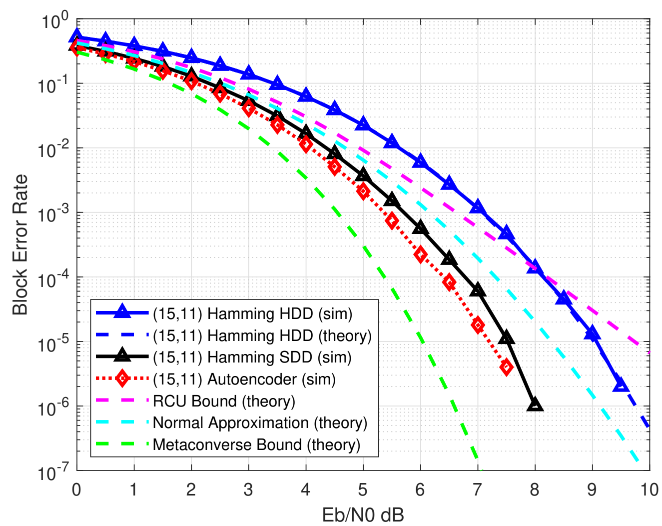

Fig. 3. BLER of the trained $(15,11,15)$ autoencoder and BPSK-modulated Hamming $(15,11)$ in an AWGN(E_b/N_0) channel.

4. AUTOENCODER FOR BGIN CHANNELS

In this section, we take a similar approach used in Section 3 and apply it to Bernoulli-Gaussian impulsive noise (BGIN) channels. The main difference here is that the autoencoder is trained separately on each Bernoulli-Gaussian probability $p_b \in \{0, 0.1, \ldots, 1\}$. This results in a family of learned autoencoders indexed by p_b. The training parameters are otherwise identical to those in Section 3.

A common approach for mitigating the effects of impulsive noise is to use clipping or blanking before demodulation [26]. With clipping, the received signal is limited to a clipping threshold, i.e.,

$$y_{\text{clipped}} = \begin{cases} y & |y| < T_c \\ \text{sign}(y) T_c & |y| \geq T_c \end{cases}$$

where T_c is the clipping threshold. Similarly, with blanking, the received signal is set to zero if it exceeds a threshold, i.e.,

$$y_{\text{blanked}} = \begin{cases} y & |y| < T_b \\ 0 & |y| \geq T_b \end{cases}$$

where T_b is the blanking threshold.

Figure 4 plots the achieved BLER of the family of trained $(7, 4, 7)$ autoencoders in $\mathrm{BGIN}(3\mathrm{dB}, -7\mathrm{dB}, p_b)$ channels. The BLER of BPSK-modulated Hamming $(7, 4)$ codes with various combinations of hard decisions, soft decisions, blanking, and clipping are also plotted for comparison. The clipping and blanking thresholds were set to $T_c = T_b = \mathrm{mean}(|y_k|)$. In this example, the autoencoder uniformly outperforms conventional coding and modulation, with or without clipping or blanking. The cyan and magenta lines represent the AWGN performance of $(7, 4)$ Hamming codes with BPSK modulation (corresponding to $p_b = 0$ and $p_b = 1$). When $p_b = 0$, channel symbols are always sent through the $\mathrm{AWGN}(7\mathrm{dB})$ channel (the less noisy channel). When $p_b = 1$, channel symbols are always sent through the $\mathrm{AWGN}(-3\mathrm{dB})$ channel (the more noisy channel). Observe that the autoencoder is more robust than hard and soft decision decoding, even with clipping or blanking, at all values of p_b.

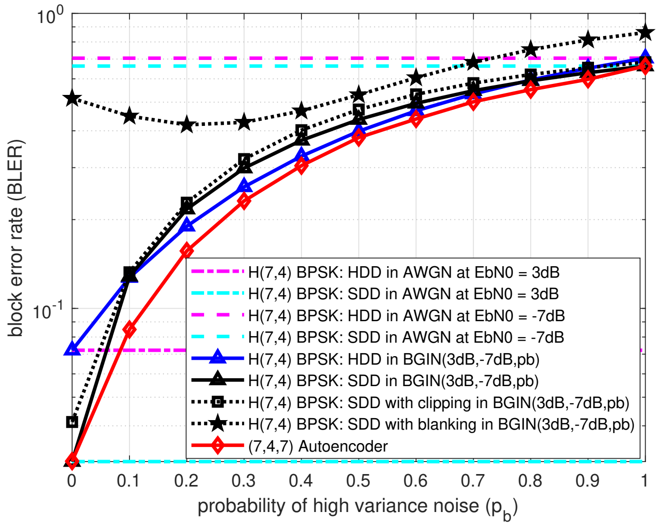

Fig. 4. BLER comparison of the family of trained $(7, 4, 7)$ autoencoders with BPSK-modulated Hamming $(7, 4)$ in $\mathrm{BGIN}(3\mathrm{dB}, -7\mathrm{dB}, p_b)$ channels.

Similarly, Figure 5 shows the BLER performance of the family of trained $(15, 11, 15)$ autoencoders compared to BPSK-modulated $(15, 11)$ Hamming codes in $\mathrm{BGIN}(3\mathrm{dB}, -7\mathrm{dB}, p_b)$ channels. Again, the achieved BLER of the autoencoder uniformly outperforms conventional coding and modulation with and without clipping and blanking (with $T_c = T_b = \mathrm{mean}(|y_k|)$). This example shows that, even with longer blocklength codes, an autoencoder trained to minimize the BLER in impulsive noise is more robust than the conventional methods at all values of p_b. The training was done on 5×10^6 examples and the parameters are otherwise identical to those of the $(15, 11, 15)$ autoencoder in Section 3.

In our tests, we observed that the adaptation rate of the autoencoder is highly dependent on the training SNR. It is important to set $\mathcal{E}_b/N_0$ such that the autoencoder observes frequent examples of correctly and incorrectly decoded blocks. This facilitates the training process. Training at high SNRs, e.g., $\mathcal{E}_b/N_0 = 8.5$ dB, does not provide enough examples of block errors and adaptation is slow in this setting. Similarly, training at a very low SNR like $\mathcal{E}_b/N_0 = -3$ dB limits the number of examples of correctly decoded blocks and results in slow adaptation.

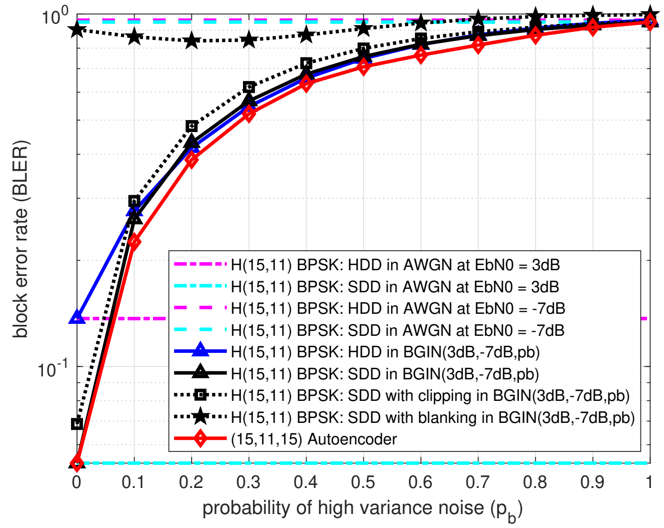

Fig. 5. BLER comparison of the family of trained $(15, 11, 15)$ autoencoders with BPSK-modulated Hamming $(15, 11)$ in $\mathrm{BGIN}(3\mathrm{dB}, -7\mathrm{dB}, p_b)$ channels.

5. CONCLUSION

In this paper, we consider the use of trained autoencoders for joint coding and modulation in the ultra-short blocklength regime with general memoryless channel models. We compare the end-to-end BLER performance of the trained autoencoders in both AWGN and impulsive noise scenarios for BPSK-modulated Hamming $(7, 4)$ and Hamming $(15, 11)$ codes. Our results demonstrate that they can learn efficient encoding, modulation and decoding functions and, in some cases, can outperform classical separately coded/modulated systems even with computationally intensive soft decision decoding. In impulsive noise, the autoencoder learns better codewords than a typical heuristic approach like clipping or blanking. These results suggest that, in situations where an autoencoder has sufficient examples for training, it can be a promising solution in challenging channels including those where a precise channel model is not available. While this paper considers memoryless channels, an interesting extension would be to investigate the use of autoencoders in the context of channels with memory, e.g., Markov-Middleton and Markov-Gaussian models. We also plan to tackle harder channel models which have no known good codes and the number of channel outputs is a random variable, such as erasure and deletion channels.

While the focus of this paper was on comparisons to Hamming codes with BPSK modulation due to page limitations, the approach described here is generally applicable to any block coding scheme and modulation format. The source code used to generate the results in this paper (as well as for Golay codes and higher order modulations) is available for download on GitHub [27].

Finally, we note that learning for an end-to-end communication system is limited by its scalability to larger blocklengths, since the autoencoder relies on one-hot-encoding and needs to be trained on all possible codewords to minimize the BLER. An interesting alternative is training on hyperspherical output spaces [28, 29] that regularize the mappings to avoid representation redundancy, thus retaining the performance while decreasing the computational complexity.

6. REFERENCES

[1] J.G. Proakis, *Digital Communications*, McGraw-Hill series in electrical and computer engineering : communications and signal processing. McGraw-Hill, 2001.

[2] G. D. Forney, "Trellis shaping," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 281–300, March 1992.

[3] G. Böcherer, F. Steiner, and P. Schulte, "Bandwidth efficient and rate-matched low-density parity-check coded modulation," *IEEE Transactions on Communications*, vol. 63, no. 12, pp. 4651–4665, Dec 2015.

[4] 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, Dec 2017.

[5] S. Dörner, S. Cammerer, J. Hoydis, and S. t. Brink, "Deep learning based communication over the air," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 132–143, Feb 2018.

[6] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *Journal of machine learning research*, vol. 11, no. Dec, pp. 3371–3408, 2010.

[7] Timothy J O'Shea, Tamoghna Roy, Nathan West, and Benjamin C Hilburn, "Physical layer communications system design over-the-air using adversarial networks," in *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018, pp. 529–532.

[8] F. A. Aoudia and J. Hoydis, "End-to-end learning of communications systems without a channel model," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, Oct 2018, pp. 298–303.

[9] Deniz Gündüz, Paul de Kerret, Nicholas D Sidiropoulos, David Gesbert, Chandra R Murthy, and Mihaela van der Schaar, "Machine learning in the air," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2184–2199, 2019.

[10] A. Felix, S. Cammerer, S. Dörner, J. Hoydis, and S. Ten Brink, "Ofdm-autoencoder for end-to-end learning of communications systems," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, June 2018, pp. 1–5.

[11] Hyeji Kim, Yihan Jiang, Sreeram Kannan, Sewoong Oh, and Pramod Viswanath, "Deepcode: Feedback codes via deep learning," in *Advances in Neural Information Processing Systems*, 2018, pp. 9458–9468.

[12] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep Learning*, MIT Press, 2016.

[13] Thokozani Shongwey, AJ Han Vinck, and Hendrik C Ferreira, "On impulse noise and its models," in *Power Line Communications and its Applications (ISPLC), 2014 18th IEEE International Symposium on*. IEEE, 2014, pp. 12–17.

[14] Dario Fertonani and Giulio Colavolpe, "On reliable communications over channels impaired by bursty impulse noise," *IEEE Transactions on Communications*, vol. 57, no. 7, pp. 2024–2030, 2009.

[15] Z. Mei, M. Johnston, S. Le Goff, and L. Chen, "Error probability analysis of m-qam on rayleigh fading channels with impulsive noise," in *2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, July 2016, pp. 1–5.

[16] Riccardo Pighi, Michele Franceschini, Gianluigi Ferrari, and Riccardo Raheli, "Fundamental performance limits of communications systems impaired by impulse noise," *IEEE Trans. on Comm.*, vol. 57, no. 1, pp. 171–182, 2009.

[17] H. Hamad and G. M. Kraidy, "Performance analysis of convolutional codes over the bernoulli-gaussian impulsive noise channel," in *2017 15th Canadian Workshop on Information Theory (CWIT)*, June 2017, pp. 1–5.

[18] K. M. Rabie and E. Alsusa, "Improving blanking/clipping based impulsive noise mitigation over powerline channels," in *2013 IEEE 24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Sep. 2013, pp. 3413–3417.

[19] S. Tseng, D. Tseng, T. Tsai, and Y. S. Han, "Robust turbo decoding in single-carrier systems over memoryless impulse noise channels," in *2016 International Conference on Advanced Technologies for Communications (ATC)*, Oct 2016, pp. 344–349.

[20] Sumrin M Kabir, Alina Mirza, and Shahzad A Sheikh, "Impulsive noise reduction method based on clipping and adaptive filters in awgn channel," *International Journal of Future Computer and Communication*, vol. 4, no. 5, pp. 341, 2015.

[21] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," 2014.

[22] David J. C. MacKay, *Information Theory, Inference & Learning Algorithms*, Cambridge University Press, New York, NY, USA, 2002.

[23] A. Martinez and A. G. i Fàbregas, "Saddlepoint approximation of random-coding bounds," in *2011 Information Theory and Applications Workshop*, Feb 2011, pp. 1–6.

[24] Y. Polyanskiy, H. V. Poor, and S. Verdu, "Channel coding rate in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[25] T. Erseghe, "Coding in the finite-blocklength regime: Bounds based on laplace integrals and their asymptotic approximations," *IEEE Transactions on Information Theory*, vol. 62, no. 12, pp. 6854–6883, Dec 2016.

[26] G. Ndo, P. Siohan, and M. Hamon, "Adaptive noise mitigation in impulsive environment: Application to power-line communications," *IEEE Transactions on Power Delivery*, vol. 25, no. 2, pp. 647–656, April 2010.

[27] Kirty Vedula, Randy Paffenroth, and D.Richard Brown III, "www.github.com/kirtyvedula/jcm-awgn-imp: First release of software associated with this paper," Oct. 2019.

[28] Weiyang Liu, Rongmei Lin, Zhen Liu, Lixin Liu, Zhiding Yu, Bo Dai, and Le Song, "Learning towards minimum hyperspherical energy," in *Advances in Neural Information Processing Systems*, 2018, pp. 6222–6233.

[29] Avradeep Bhowmik, Nathan Liu, Erheng Zhong, Badri Bhaskar, and Suju Rajan, "Geometry aware mappings for high dimensional sparse factors," in *Artificial Intelligence and Statistics*, 2016, pp. 455–463.