DEEP LEARNING FOR FRAME ERROR PROBABILITY PREDICTION IN BICM-OFDM SYSTEMS

Vidit Saxena[1,2], Joakim Jaldén[1], Mats Bengtsson[1], and Hugo Tullberg[2]

[1]Department of Information Science and Engineering, KTH, Stockholm
[2]Ericsson Research, Stockholm

ABSTRACT

In the context of wireless communications, we propose a deep learning approach to learn the mapping from the instantaneous state of a frequency selective fading channel to the corresponding frame error probability (FEP) for an arbitrary set of transmission parameters. We propose an abstract model of a bit interleaved coded modulation (BICM) orthogonal frequency division multiplexing (OFDM) link chain and show that the maximum likelihood (ML) estimator of the model parameters estimates the true FEP distribution. Further, we exploit deep neural networks as a general purpose tool to implement our model and propose a training scheme for which, even while training with the binary frame error events (i.e., ACKs / NACKs), the network outputs converge to the FEP conditioned on the input channel state. We provide simulation results that demonstrate gains in the FEP prediction accuracy with our approach as compared to the traditional effective exponential SIR metric (EESM) approach for a range of channel code rates, and show that these gains can be exploited to increase the link throughput.

Index Terms— FEP, BICM-OFDM, Deep Learning, Neural Networks, Link Adaptation.

1. INTRODUCTION

The efficiency of a radio link depends on its ability to adapt to the stochastic radio channel conditions that typically vary over time (i.e., fading) as well as over the signal bandwidth (i.e., frequency selectivity). Practical radio systems perform this so-called "link adaptation" by selecting the optimal transmission parameters in each frame that fulfil some criteria related to, e.g., target error rates, throughput, or latency etc. [1]. In this paper, we investigate the problem of predicting the frame error probability (FEP) for an estimated channel state in bit-interleaved coded modulation (BICM) orthogonal frequency division multiplexing (OFDM) systems [2] [3]. Owing to their flexibility and performance, BICM-OFDM systems have been widely adopted by most of the modern radio air interfaces including those for local wireless area networks (e.g., WiFi) and for cellular communication such as Long Term Evolution for 4G, and recently, New Radio for 5G [4].

In general for frequency selective channels, it is intractable to compute the FEP conditioned on the frame channel state characterized by the received per-subcarrier signal to interference and noise ratios (SINRs). Therefore, several approximate techniques for FEP prediction have been developed that compress the per-subcarrier SINRs to an approximate effective scalar metric, which is mapped to pre-computed FEP values stored as lookup tables [5] [6] [7]. However, these techniques assume ideal channel coding performance and do not take into account practical system impairments. Further, the choice of a suitable compression function can be somewhat arbitrary and has been empirically shown to have a significant impact on the FEP prediction performance [8].

As an alternative to the effective SINR approach, supervised learning techniques that map the per-subcarrier SINR vector for each frame to the corresponding FEP have been proposed [9] [10]. During the traning phase of these techniques, the model parameters are selected to minimize the mean cost between the model output for several frame channel state vectors and their Monte Carlo simulated FEPs. With sufficient training, these techniques have been shown to improve the realized link throughput of BICM-OFDM systems compared to the effective SINR approach. However, the proposed supervised learning techniques are limited by the accuracy of the simulated training datasets and additionally do not provide any insight into the optimality of the trained models.

In this paper, we cast the FEP prediction problem as a probabilistic binary classification task, where the classes correspond to frame error and success events (i.e., NACKs and ACKs) respectively, and make the following three main contributions: (i) We propose an abstract model of the BICM-OFDM link chain where the observations are the frame channel states and their binary frame error events and show that, in the limit of infinite training samples, the maximum likelihood (ML) estimator of the model parameters estimates the true FEP distribution, (ii) We use this model to develop a supervised learning approach for FEP prediction based on deep neural networks, where the training phase requires only the observed channel states and binary frame error events, thus mitigating the need for simulations to measure the FEP, and (iii) We provide simulation results to show that our approach improves the FEP prediction accuracy, and consequently the link throughput, compared to the well-studied exponential effective SIR metric (EESM) approach.

The rest of this paper is organized as follows: In Sec. 2, we describe a typical BICM-OFDM wireless communication link and propose an abstract system model along with an ML estimator of the model parameters. Next in Sec. 3, we summarize the EESM approach for FEP prediction and introduce our deep learning approach. Next in Sec. 4 we present simulation results for the FEP prediction performance as well as the realized throughput for our two considered approaches and finally in Sec. 5, we conclude the paper.

2. BICM-OFDM SYSTEM AND ML ESTIMATION

2.1. System Model

We consider a BICM-OFDM link chain similar to the LTE downlink and illustrate its block diagram in Fig. 1 [11]. Here, a "transport block", $\boldsymbol{b} = (b_1, \ldots, b_T)$ of information bits is first encoded

This work was partially supported by the Wallenberg Autonomous Systems and Software Program (WASP).

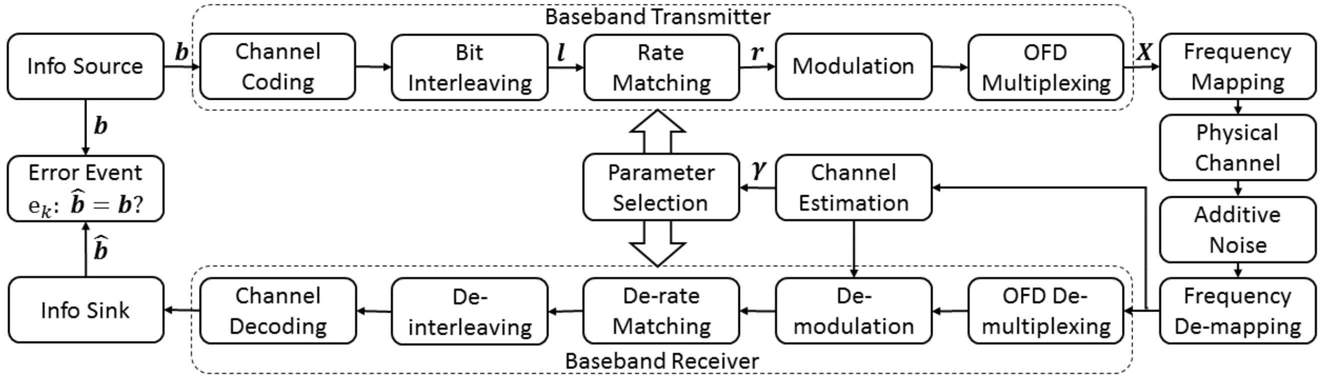

Fig. 1. Block diagram of the BICM-OFDM link chain considered in this paper. The parameter selection module exploits knowledge of the instantaneous channel state to select one out of several possible transmission parameter configurations in each frame.

by a channel encoder and subsequently bit-interleaved by a random interleaver to generate the bit sequence $\boldsymbol{l} = (l_1, \ldots, l_L)$. The interleaved bits are then used to generate the "rate-matched" bit sequence $\boldsymbol{r} = (r_1, \ldots, r_{MSJ})$ according to $r_i = l_{i \bmod L}, i = 1, \ldots, MSJ$, where M is the number of transmission subcarriers, S is the number of frame OFDM symbols, and J is the modulation order. The channel code rate is therefore $\mathcal{R} = T/MSJ$. The rate matched bits are mapped onto MS modulated symbols by a labeling function that assigns one out of 2^J complex-valued constellation symbols to each J-tuple of bits. Finally, a length-M IFFT operation is applied on each group of M modulated symbols to generate the frame OFDM symbols $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_S)$ and mapped onto physical resources.

The frame OFDM symbols are transmitted over the physical channel resulting in the received signal $\boldsymbol{y}_s = \boldsymbol{h} \odot \boldsymbol{x}_s + \boldsymbol{g}_s, s = 1, \ldots, S$, where $\odot$ denotes the Hadamard product, $\boldsymbol{h}$ is the complex-valued vector of channel coefficients in the frequency domain, and $\boldsymbol{g}_s \sim \mathcal{N}^{M\times 1}(0, \sigma^2)$ is i.i.d. noise. We assume that the channel vector remains constant for the frame OFDM symbols (i.e., the channel is block fading). At the receiver, each received OFDM symbol $\boldsymbol{y}_s$ is multiplied by the elementwise inverse of the estimated frequency-domain channel vector, followed by a length-M FFT operation for OFDM de-multiplexing. The de-multiplexed symbols are then mapped onto soft values through an inverse labeling operation, de-interleaved, and decoded by the channel decoder to generate the reconstructed bit sequence $\hat{\boldsymbol{b}} = (\hat{b}_1, \ldots, \hat{b}_T)$. We define the binary frame error event at the receiver as

$$e = \begin{cases} 0 & \text{if} \quad \hat{\boldsymbol{b}} = \boldsymbol{b} \\ 1 & \text{if} \quad \hat{\boldsymbol{b}} \neq \boldsymbol{b} \end{cases}. \tag{1}$$

2.2. ML Estimation

The BICM-OFDM link chain described above can be approximated as a stochastic non-linear function that generates a frame error event with an unknown probability distribution for the frame channel state and a particular choice of transmission parameters. In Fig. 2, we illustrate an abstract model of the BICM-OFDM link chain, which is parameterized by the model parameters $\boldsymbol{\theta}$ and maps the observed channel state characterized by the received per-subcarrier channel SINRs, $\boldsymbol{\gamma} = (\gamma_1 \ldots, \gamma_M)$, to the observed frame error event, .i.e.,

$$P_{E_k|\boldsymbol{\Gamma}}(e_k|\boldsymbol{\gamma};\boldsymbol{\theta}) = \rho_k^{e_k}(1-\rho_k)^{1-e_k}, \tag{2}$$

where the $k \in 1, \ldots, K$ denotes the k^{th} transmission parameter configuration. Here, $\rho_k = \rho_k(\boldsymbol{\gamma};\boldsymbol{\theta}) = P_{E_k|\boldsymbol{\Gamma}}(E_k = 1|\boldsymbol{\gamma};\boldsymbol{\theta})$ is the conditional frame error probability (FEP) . In the rest of this section, we show that the ML estimator of the model parameters asymptotically estimates the true conditional FEPs.

The ML estimator [12] of the model parameters for $n = 1, \ldots, N$ frame realizations is defined as

$$\begin{aligned} \hat{\boldsymbol{\theta}}^{\text{ML}} &= \arg\max_{\hat{\boldsymbol{\theta}}} \sum_{k=1}^{K} \prod_{n=1}^{N} P_{E_k,\Gamma}(e_k^n, \boldsymbol{\gamma}^n; \hat{\boldsymbol{\theta}}) \\ &= \arg\max_{\hat{\boldsymbol{\theta}}} \int \sum_{k=1}^{K}\sum_{n=1}^{N} \ln P_{E_k|\Gamma}(e_k^n|\boldsymbol{\gamma}^n;\hat{\boldsymbol{\theta}}) P_{\boldsymbol{\Gamma}}(\boldsymbol{\gamma}) d\boldsymbol{\gamma} \\ &\triangleq \arg\max_{\hat{\boldsymbol{\theta}}} \mathcal{C}(\hat{\boldsymbol{\theta}}), \text{ where} \end{aligned} \tag{3}$$

$$\mathcal{C}(\hat{\boldsymbol{\theta}}) = \sum_{k=1}^{K} \left(\frac{1}{N} \sum_{n=1}^{N} \ln P_{E_k|\Gamma}(e_k^n|\boldsymbol{\gamma}^n;\hat{\boldsymbol{\theta}}) \right) \tag{4}$$

is the cost function to be maximized, and we have used the fact that the channel state is independent of the model. In the limit of infinite training samples, it follows by the law of large numbers that

$$\mathcal{C}(\hat{\boldsymbol{\theta}}) \xrightarrow{N\to\infty} \sum_{k=1}^{K} E\{\ln P(E_k|\boldsymbol{\Gamma};\hat{\boldsymbol{\theta}})\} \tag{5}$$

where $P(E_k|\boldsymbol{\Gamma};\hat{\boldsymbol{\theta}}) \triangleq P_{E_k|\boldsymbol{\Gamma}}(E_k|\boldsymbol{\Gamma};\hat{\boldsymbol{\theta}})$ for brevity, and where the expectation is taken over $P(E_k, \boldsymbol{\Gamma};\boldsymbol{\theta})$. We now subtract and add the true pmf to the r.h.s. of Eq. (5) to obtain

$$\begin{aligned} \mathcal{C}(\hat{\boldsymbol{\theta}}) &= \sum_{k=1}^{K} E\{\ln P(E_k|\boldsymbol{\Gamma};\hat{\boldsymbol{\theta}}) - \ln P(E_k|\boldsymbol{\Gamma};\boldsymbol{\theta}) + \ln P(E_k|\boldsymbol{\Gamma};\boldsymbol{\theta})\} \\ &= -\sum_{k=1}^{K} E\left\{\ln \frac{P(E_k|\boldsymbol{\Gamma};\boldsymbol{\theta})}{P(E_k|\boldsymbol{\Gamma};\hat{\boldsymbol{\theta}})}\right\} + \sum_{k=1}^{K} E\left\{\ln P(E_k|\boldsymbol{\Gamma};\boldsymbol{\theta})\right\}. \end{aligned} \tag{6}$$

The second term in (6) is independent of the argument to be maximized. Further we observe that by multiplying and dividing the first term with probability distribution $P_{\boldsymbol{\Gamma}}(\boldsymbol{\gamma})$, we obtain

$$\sum_{k=1}^{K} E\left\{\ln \frac{P(E_k|\boldsymbol{\Gamma};\boldsymbol{\theta})P(\boldsymbol{\Gamma})}{P(E_k|\boldsymbol{\Gamma};\hat{\boldsymbol{\theta}})P(\boldsymbol{\Gamma})}\right\} = \sum_{k=1}^{K} E\left\{\ln \frac{P(E_k,\boldsymbol{\Gamma};\boldsymbol{\theta})}{P(E_k,\boldsymbol{\Gamma};\hat{\boldsymbol{\theta}})}\right\} = \sum_{k=1}^{K} \text{KL}\left\{P(E_k|\boldsymbol{\Gamma};\boldsymbol{\theta})||P(E_k|\boldsymbol{\Gamma};\hat{\boldsymbol{\theta}})\right\}, \tag{7}$$

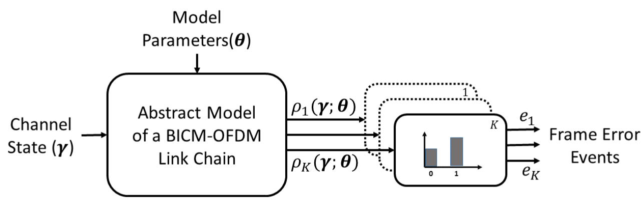

Fig. 2. Abtract model of a BICM-OFDM link chain that maps the observed channel state to the frame error events for $k = 1, \ldots, K$ transmission parameter configurations.

where $\mathrm{KL}(\cdot||\cdot)$ is the Kullback-Leibler divergence (KLD) between the true and estimated pmfs. Given that the KLD is non-negative, and equal to zero if and only if $P(E_k|\boldsymbol{\Gamma};\boldsymbol{\theta}) = P(E_k|\boldsymbol{\Gamma};\hat{\boldsymbol{\theta}})$, it follows that the ML estimator converges to the true FEP distribution in the limit of large N. Note however that $P(E_k|\boldsymbol{\Gamma};\boldsymbol{\theta}) = P(E_k|\boldsymbol{\Gamma};\hat{\boldsymbol{\theta}})$ does not necessarily imply that $\hat{\boldsymbol{\theta}}^{\mathrm{ML}} = \boldsymbol{\theta}$ as the ML estimate of $\boldsymbol{\theta}$ may be non-unique.

3. FEP PREDICTION TECHNIQUES

3.1. Effective SINR Approach

In this subsection we outline the EESM approach, where the the channel state characterized by the per-subcarrier SINRs is compressed to a scalar "effective" SINR for an equivalent AWGN channel. The FEP for the k^{th} transmission parameter configuration is then predicted to be

$$\hat{\rho}_k^{\mathrm{EESM}}(\boldsymbol{\gamma}) = \rho_k^{\mathrm{AWGN}}(g_k(\boldsymbol{\gamma})), \tag{8}$$

$$\text{where } g_k(\boldsymbol{\gamma}) = -\beta_k \log\left(\frac{1}{M}\sum_{m=1}^{M}\exp\left(-\frac{\gamma_m}{\beta_k}\right)\right) \tag{9}$$

is the EESM for channel state $\boldsymbol{\gamma}$, and β_k is a tunable parameter. For the frequency selective channels commonly observed in practical systems, $g_k(\boldsymbol{\gamma})$ amounts to a lossy compression of the channel state vector, since the original channel state can no longer be recovered. The FEP for the equivalent AWGN channel, $\rho_k^{\mathrm{AWGN}}(g_k(\boldsymbol{\gamma}))$, is obtained by interpolating between several Monte Carlo simulated FEP values for the AWGN channel. The optimal β_k minmizes the Euclidean distance between the predicted FEP and observed frame errors for $n = 1, \ldots, N$ training frames, i.e.,

$$\beta_k^{\mathrm{opt}} = \arg\min_{\beta_k} \sum_{n=1}^{N} |\hat{\rho}_k^{\mathrm{EESM,\,n}} - e_k^n|^2. \tag{10}$$

For large N, estimating β_k^{opt} in this manner is equivalent to the traditional approach that minimizes the mean squared cost between the predicted FEPs and and the measured FEPs obtained through Monte Carlo simulations for the training frames [8].

3.2. Deep Learning Approach

The EESM approach described earlier relies on a scalar approximation of the channel state, which is obtained through a lossy compression and therefore does not guarantee optimality of the the corresponding FEP prediction. In this subsection we describe our approach for FEP prediction based on deep neural networks, which discriminatively learns the mapping between the (uncompressed) channel state vector and the corresponding FEPs for multiple transmission parameter configurations.

Neural networks have long been known as a powerful tool for approximating a wide range of highly non-linear functions, however, their acceptance for implementation in practical systems has been limited by an insufficient understanding of the models that they learn from training data. Although a complete understanding of neural networks is still a topic of active research, several recent breakthroughs related to deep neural networks coupled with cheap computational power have led to drastic performance improvements for several challenging problems [13]. In this paper, we consider the fully connected $L-$layered feedforward neural network illustrated in Fig. 3, for which the output of the l^{th} "hidden layer" with dimension d_l can be described as

$$\boldsymbol{\eta}^{(l)} = \boldsymbol{\phi}^{(l)}\left(\boldsymbol{W}^{(l)}\boldsymbol{\eta}^{(l-1)} + \boldsymbol{b}^{(l)}\right), \tag{11}$$

where $\boldsymbol{W}^{(l)}$ is the trainable $d_{l-1} \times d_l$ weight matrix, $\boldsymbol{b}^{(l)}$ is the trainable $d_l \times 1$ bias vector, and $\boldsymbol{\phi}^{(l)}$ is a fixed non-linear "activation" function. By simply substituting the system parameters $\boldsymbol{\theta}$ with neural network weights and biases, we allow the neural network to learn the set of mappings $\rho_k(\boldsymbol{\gamma};\hat{\boldsymbol{\theta}})$ for $k = 1, \ldots, K$ from data.

It has been shown previously that an ordering of the per-subcarrier SINRs can be used to sufficiently parameterize the frame error rate while reducing the training requirements [9]. Therefore in this paper, we use the sorted per-subcarrier SINR vector, $\widetilde{\boldsymbol{\gamma}}$, as the input to the network, i.e., $\boldsymbol{\eta}^{(0)} \triangleq \widetilde{\boldsymbol{\gamma}} = (\widetilde{\gamma}_1, \ldots, \widetilde{\gamma}_M)$. The activation function for the each of the non-output layers can be any continuously differentiable non-linear function within some practical constraints [13]. For the output layer, we choose sigmoid activation function $\boldsymbol{\phi}^{(L)}(x) = 1/(1+e^{-x})$ and interpret the network outputs as the predicted FEPs, i.e., $\boldsymbol{\eta}^{(L)} \triangleq \hat{\boldsymbol{\rho}}^{\mathrm{NN}} = (\hat{\rho}_1^{\mathrm{NN}}, \ldots, \hat{\rho}_K^{\mathrm{NN}})$. We choose the cross-entropy loss between the network outputs and the frame error events as the cost function, i.e.,

$$\mathcal{C}(\hat{\boldsymbol{\theta}}) = \frac{1}{K}\sum_{k=1}^{K} e_k \ln \hat{\rho}_k^{\mathrm{NN}} + (1-e_k)\ln(1-\hat{\rho}_k^{\mathrm{NN}}) \tag{12}$$

$$= \frac{1}{K}\sum_{k=1}^{K} \ln P_{E_k|\boldsymbol{\Gamma}}(e_k|\boldsymbol{\gamma};\hat{\boldsymbol{\theta}}), \tag{13}$$

where the latter expression is obtained using Eq. (2). By observing the direct correspondence between Eqs. (4) and (13), we observe that minimizing the neural network cost function over $n = 1, \ldots, N$ training frames is equivalent to ML estimation of the neural network parameters, which we have shown to estimate the true FEPs.

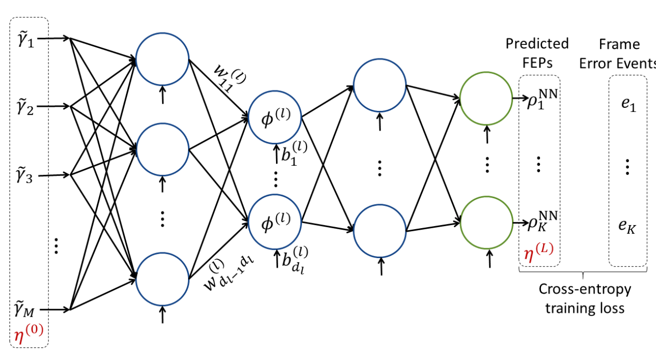

Fig. 3. Neural network layout that maps the channel state characterized by the sorted per-subcarrier SINR values to the FEP for each of the K transmission parameter configurations.

It is crucial to point out that there is no guarantee that the neural network trained using stochastic gradient decent will converge to the ML estimate of the network parameters, however, our simulation results indicate that the neural network does indeed provide good estimates of $\rho_k(\boldsymbol{\gamma};\hat{\boldsymbol{\theta}})$. Further, the result obtained above is equivalent to the conclusions of a previous analysis of neural networks [14]. However, we believe that it is instructive to demonstrate this result in the context of ML estimation as well.

3.3. FEP Prediction for Throughput Maxmization

In this paper, we study the selection of the optimal channel code rate $\mathcal{R}_k, k \in \{1,\ldots,K\}$ that maximizes the link throughput over a fading channel. Therefore for each FEP prediction approach, we select the channel code rate that maximizes the predicted expected throughput in that frame, i.e., $k^{\text{EESM}} = \arg\max_k T_k(1-\rho_k^{\text{EESM}})$, and $k^{\text{NN}} = \arg\max_k T_k(1-\rho_k^{\text{NN}})$, where $T_k = MSJ\mathcal{R}_k$ is the number of transmitted information bits when the k^{th} channel code rate is selected. The realized throughput over N evaluted frames is therefore $\mathcal{T}^{\text{EESM}} = \frac{1}{N}\sum_n T^n_{k^{\text{EESM}}}(1-e^n_{k^{\text{EESM}}})$ and $\mathcal{T}^{\text{NN}} = \frac{1}{N}\sum_n T^n_{k^{\text{NN}}}(1-e^n_{k^{\text{NN}}})$ respectively, where e_k^n denotes the actual frame error event for the k^{th} channel code rate in the n^{th} frame.

4. NUMERICAL RESULTS

In this section we provide simulation results for the FEP prediction accuracy and the achieved link throughput for our proposed deep learning approach and contrast it with the EESM approach performance. The simulation parameters are listed in Table 1. We use open source Python signal processing and communication libraries to foster reproducibility of the demonstrated results described in this section, and utilize Tensorflow for the neural network implementation and evaluations [15] [16].

We assume perfect knowledge of the channel at the transmitter as well as the receiver, i.e., $\boldsymbol{\gamma} = |\boldsymbol{h}|^2/\sigma^2$. The neural network comprises 3 hidden layers with dimensions $[60, 10, 60]$ respectively and each hidden layer employs a Rectified Linear Unit (ReLU) activation function [13]. The training datasets for EESM and neural network approaches are generated using 10^4 frames for each channel code rate. The test dataset for throughput maximization comprises 10^3 realizations for 10 evenly spaced long term average SINR values in the range $[-10, 20]$ dB.

Table 1. Simulation Parameters

Simulation Parameter	Value
Channel Model	Extended Pedestrian A [17]
Max. Doppler Spread	3 Hz
Number of OFDM Symbols (S)	12
Number of Subcarriers (M)	600
Channel Coding	Turbo + Repetition
Channel Code Rates ($\mathcal{R}_k$)	$[0.01, 0.02\ldots, 0.30]$

We train the neural network parameters iteratively be employing an ADAM optimizer [13]. The Root Mean Square Error (RMSE) of the FEP prediction performance versus the number of neural network training steps is shown in Fig. 4. We observe that the neural network learns to improve the FEP prediction by iteratively training its parameters, and outperforms the EESM approach after a few iterations. The throughput performance with the EESM and our deep learning approach is shown in Fig. 5. We observe that our approach increases the throughput compared to the EESM approach.

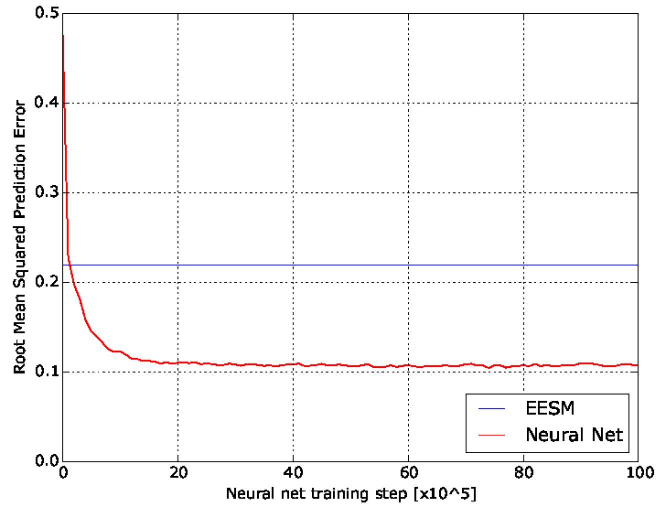

Fig. 4. RMSE of the FEP prediction error for EESM (blue, solid) and deep learning (red, solid) approaches for the simulation setup in Table 1. The neural network iteratively learns model parameters that improve its FEP prediction accuracy compared to EESM approach.

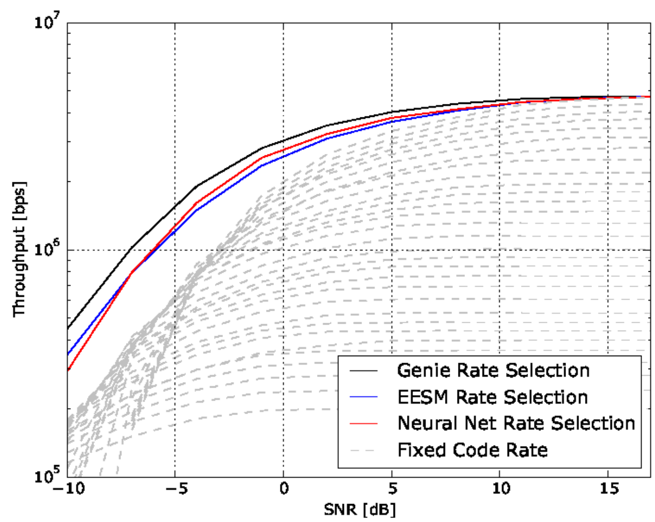

Fig. 5. Link throughput for EESM (blue, solid) and deep learning (red, solid) based rate selection that maximizes the expected throughput in each frame. The upper bound "Genie" curve (black, solid) is obtained by simulating each channel code code rate for each frame and picking the largest transport block that is successful. The lower bound "Fixed Code Rate" curves (gray, dashed) are obtained by fixing the channel code rate over the entire test dataset.

5. CONCLUSIONS

In this paper, we have proposed a deep learning approach for FEP prediction that learns the mapping between the frame channel state characterized by the per-subcarrier SINRs and the FEPs for arbitrary transmission parameter configurations. Further by utilizing a training scheme that relies only on the observed channel state and the binary frame error events, our approach is shown to improve the FEP prediction accuracy and consequently increase the link throughput.

6. REFERENCES

[1] A. Molisch, *Wireless Communications*. Wiley, 2010.

[2] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Transactions on Information Theory*, vol. 44, no. 3, pp. 927–946, May 1998.

[3] D. Agrawal, V. Tarokh, A. Naguib, and N. Seshadri, "Space-time coded OFDM for high data-rate wireless communication over wideband channels," in *Vehicular Technology Conference, 1998. VTC 98. 48th IEEE*, vol. 3, May 1998, pp. 2232–2236 vol.3.

[4] A. Zaidi, R. Baldemair, M. Andersson, S. Faxér, V.-M. Casés, and Z. Wang, "Designing for the future: the 5G NR physical layer," *Ericsson Technology Review*, July 2017.

[5] S. Nanda and K. M. Rege, "Frame error rates for convolutional codes on fading channels and the concept of effective E_b/N_0," *IEEE Transactions on Vehicular Technology*, vol. 47, no. 4, pp. 1245–1250, Nov 1998.

[6] Y. W. Blankenship, P. J. Sartori, B. K. Classon, V. Desai, and K. L. Baum, "Link error prediction methods for multicarrier systems," in *IEEE 60th Vehicular Technology Conference, 2004. VTC2004-Fall. 2004*, vol. 6, Sept 2004, pp. 4175–4179.

[7] Ericsson, "Effective SNR mapping for modelling frame error rates in multiple-state channels," Tech. Rep. 3GPP2-C30-20030429-010, April 2003.

[8] K. Brueninghaus, D. Astely, T. Salzer, S. Visuri, A. Alexiou, S. Karger, and G. A. Seraji, "Link performance models for system level simulations of broadband radio access systems," in *2005 IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 4, Sept 2005, pp. 2306–2311.

[9] R. C. Daniels, C. M. Caramanis, and R. W. Heath, "Adaptation in Convolutionally Coded MIMO-OFDM Wireless Systems Through Supervised Learning and SNR Ordering," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 114–126, Jan 2010.

[10] S. Yun and C. Caramanis, "Multiclass support vector machines for adaptation in MIMO-OFDM wireless systems," in *2009 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Sept 2009, pp. 1145–1152.

[11] 3rd Generation Partnership Project, "Evolved Universal Terrestrial Radio Access (E-UTRA); Physical layer procedures," Tech. Rep. 36.213 v12.3.0, Sep. 2016.

[12] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice Hall PTR, 1993.

[13] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[14] M. D. Richard and R. P. Lippmann, "Neural Network Classifiers Estimate Bayesian a Posteriori Probabilities," *Neural Computation*, vol. 3, no. 4, pp. 461–483, 1991.

[15] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[16] "py-itpp: Python library for signal processing and communications research." 2016, software available on Github. [Online]. Available: https://github.com/vidits-kth/py-itpp

[17] 3rd Generation Partnership Project, "Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) radio transmission and reception," Tech. Rep. 36.104 v12.3.0, Sep. 2016.