Geometric Constellation Shaping with Low-complexity Demappers for Wiener Phase-noise Channels

Andrej Rode[*] and Laurent Schmalen

Communications Engineering Lab (CEL), Karlsruhe Institute of Technology, 76187 Karlsruhe, Germany

[*]`rode@kit.edu`

Abstract: We show that separating the in-phase and quadrature component in optimized, machine-learning based demappers of optical communications systems with geometric constellation shaping reduces the required computational complexity whilst retaining their good performance. © 2023 The Author(s)

1. Introduction

In recent years, deep learning and auto-encoders for end-to-end (E2E) optimization of optical communication systems [1, 2] have gained much attention. Specifically, geometric constellation shaping (GCS) for the Wiener phase noise channel is a promising approach to improve spectral efficiency [3, 4]. Often, blocks of the digital signal processing (DSP) chain have been left out of the optimization via auto-encoders, either due to an increase in computational complexity or problems which they introduce in the differentiability of the channel [5], required for optimization. One such DSP block is the blind phase search (BPS) [6], which is a popular algorithm for blind, feed-forward, carrier phase synchronization (CPS) in high-rate coherent optical communication receivers. In [7], the authors implemented a differentiable BPS to enable a fully differentiable channel model for the optimization of GCS in the presence of Wiener phase noise.

Previous works [4, 7–9] successfully applied GCS to improve the spectral efficiency and robustness of optical communication systems with BPS. To harness the full potential of a neural receiver, previous works use neural network (NN)-based demappers with multiple fully connected layers. This allows the system to train the neural network to fully adapt to the channel. Due to the high number of weights and, subsequently, a high number of arithmetic operations per received symbol, such NN-based demappers add computational complexity to the receiver. In a related work, it has been explored to reduce the computational complexity by limiting the NN-based demapper to a single hidden layer [10]. In this work, we present an approach to reduce the computational complexity by separating the in-phase and quadrature components at the NN-based demapper and reducing the size of the demapper NNs in both width, and depth. We compare our approach to a simple reduction of the NN-based demapper in width and size. The separation of the in-phase and quadrature components follows the implementation of demappers for classical square QAM.
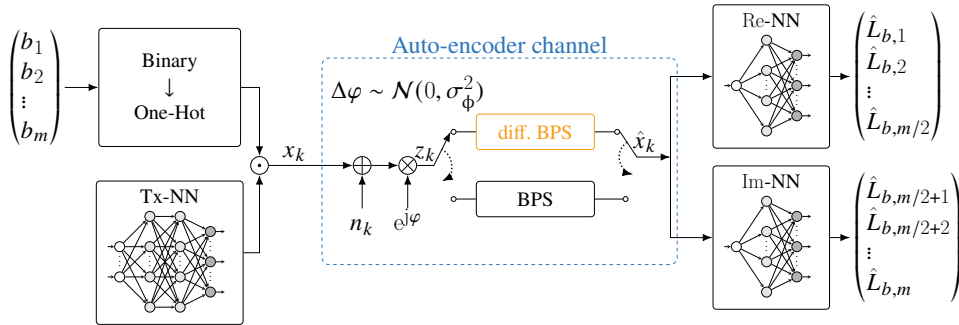
2. System Model

Fig. 1: Block diagram of the bitwise auto-encoder system with a separated demapper NN

In our proposed approach, we model an E2E optical transmission system at the symbol rate. The transmitter is implemented as an array of trainable weights for the in-phase and quadrature component, which is the Tx-NN in the block diagram of the system in Fig. 1. Following our previous works [7, 9], we implement GCS following the bitwise auto-encoder approach with bit interleaved coded modulation (BICM) [11]. In the bitwise auto-encoder, bit vectors of length m are mapped to a complex-valued constellation of size $M = 2^m$. This is accomplished by taking the dot-product between the one-hot vector and the vector of complex constellation points at the output of

the Tx-NN. The complex transmit symbols are impaired by additive white Gaussian noise (AWGN) with standard deviation σ_n and Wiener phase noise with the standard deviation σ_ϕ of the process increments. Both values are selected to represent a specific signal-to-noise ratio (SNR) and laser linewidth at the given symbol rate. The impaired and distorted complex received symbols are then sent through a CPS, which is implemented by means of BPS. During training, the operation of BPS is approximated by the differentiable BPS [7] with a temperature parameter, which is gradually reduced from 1.0 to 0.001 to approximate the non-differentiable BPS more closely at the end of the training. After the BPS, the received and corrected symbols $\hat{x}_k$ are separated in in-phase and quadrature and passed through individual NNs, Re-NN and Im-NN, each with one hidden layer, as depicted in Fig. 1. Each NN then returns $m/2$ log likelikood ratios (LLRs), which are used to calculate the loss function, the binary cross entropy (BCE). The BCE is used as a loss function to maximize the bitwise mutual information (BMI) [11]. To validate the E2E performance in the context of an implementable system, the non-differentiable BPS algorithm is used in the validation runs. Parameters of the BPS are selected as $N_{\text{angles}} = 60$ test angles and a window size of $N = 120$ for averaging.

3. Complexity Assessment of Neural Network-based Demappers

To compare the complexity of our proposed approach, we compare the computational complexity of classical calculation of LLRs with its computation via neural networks. As pointed out in [10], soft demapping to obtain LLRs can be performed with a log-MAP demapper with AWGN assumption or with an NN, formulated respectively with

$$L_i(y) = \ln\left(\frac{\sum_{x \in \mathcal{X}_i^0} \exp\left(\frac{|y-x|^2}{N_0}\right)}{\sum_{x \in \mathcal{X}_i^1} \exp\left(\frac{|y-x|^2}{N_0}\right)}\right) \quad \text{and} \quad \boldsymbol{L}(y) = \left(\tau\left(\boldsymbol{y}^T \boldsymbol{W}_1 + \boldsymbol{b}_1\right)\right)^T \boldsymbol{W}_2 + \boldsymbol{b}_2. \tag{1}$$

$\mathcal{X}_i^b$ is the set of symbols with a binary label that takes on value b at the ith bit position. The computational complexity for calculating an LLR vector for an M-quadrature amplitude modulation (QAM) constellation consists of calculating M complex subtractions and M complex multiplications. Afterwards, $2m = 2\log_2(M)$ sums with $M/2$ elements each have to be accumulated and the exponential has to be calculated with another m divisions and calculation of $\ln$ following these operations. The complexity of NNs can be assessed in terms of the vector-matrix multiplication needed for each layer. Each layer in an NN needs nm multiplications, nm additions, and m executions of the non-linear activation function τ, where n is the input width and m is the output width. For an NN-based demapper with input width k, ℓ hidden layers and m outputs, the number of multiplications $L_{\text{NN,mult}}$ required is $L_{\text{NN,mult}} = kn + (\ell - 1)n + nm$. We compare the computational complexity between the Gaussian demapper and the NN-based demapper via the number of real-valued multiplications required to calculate the LLRs per received symbol. The Gaussian demapper in (1) needs $4M$ real-valued multiplications per received symbol. If we relate this to the NN-based demapper in (1), which operates on a complex input and a single hidden layer of size n, and solve for n we obtain $n = \frac{4(2^m)}{m+2}$. We assume that one complex multiplication can be expressed with 4 real-valued multiplications. For $m = 6$, a neural network with a single hidden layer of width $n = 32$ needs a similar number of real-valued multiplications compared to a soft demapper with Gaussian noise assumption. For two NNs processing the received symbol separately in in-phase and quadrature components, the number of real-valued multiplications is the same for the input and the output layer compared to a single NN with complex input.

4. Results

When constraining the demappers by separation to in-phase and quadrature components we obtain GCS-formats resembling classical square QAM constellations. One half of the bit vector is mapped to amplitudes in the in-phase and the other half to the amplitudes in quadrature. Even though the constellation resembles square QAM, we obtain constellations that are optimized to aid the BPS algorithm in recovering and correcting the carrier phase without the aid of any pilot sequence. This can be inferred from the non-uniform distances between the amplitudes and asymmetrical placement of corner symbols in the constellation diagram in Fig 2-a). At the same time, the demappers for in-phase and quadrature components are constrained to only one hidden layer and 8 weights. Compared to other constellation diagrams trained with low-complexity demappers which do not treat in-phase and quadrature separately, as depicted in Fig. 2-b) and c), we observe a better separation of constellation points in Fig. 2-a). In Fig. 3 .we compare the performance of the constellations from Fig. 2 and their corresponding neural demappers in terms of BMI. All systems have been trained on the same SNR of 17 dB and a laser linewidth of $\Delta\nu = 100$ kHz. In Fig. 3-a), the performance is depicted for a fixed linewidth of $\Delta\nu = 100$ kHz and varying SNR. In Fig. 3-b), the performance is shown if the SNR is constant at 17 dB and the laser linewidth is changed. In both plots, the constellation with the separated demapper is able show a similar performance compared to the constellation and neural demapper presented in [7], which uses 3 hidden layers with 128 nodes per layer. In comparison, the separated
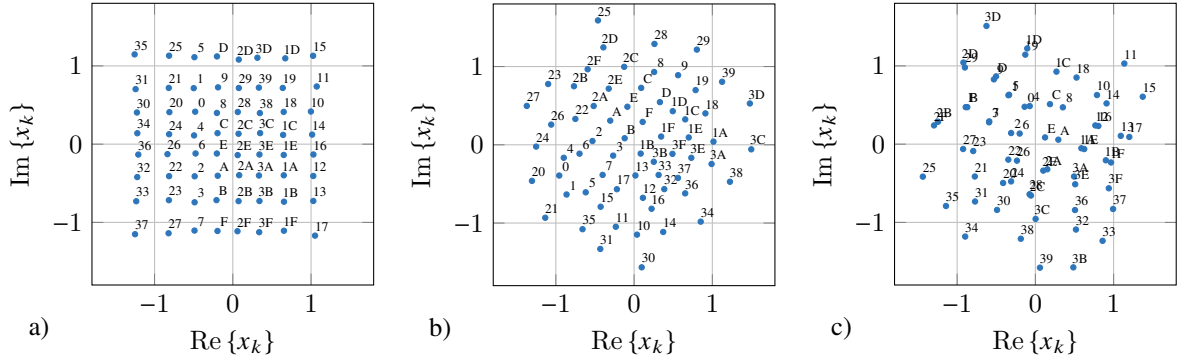
Fig. 2: Constellation diagrams for $M = 64$, 17 dB SNR and laser linewidth 100 kHz, a) is trained with a separated low-complexity demapper with $n = 8$, b) is trained with a low-complexity demapper with $n = 16$ and c) is trained with a low-complexity demapper with $n = 8$
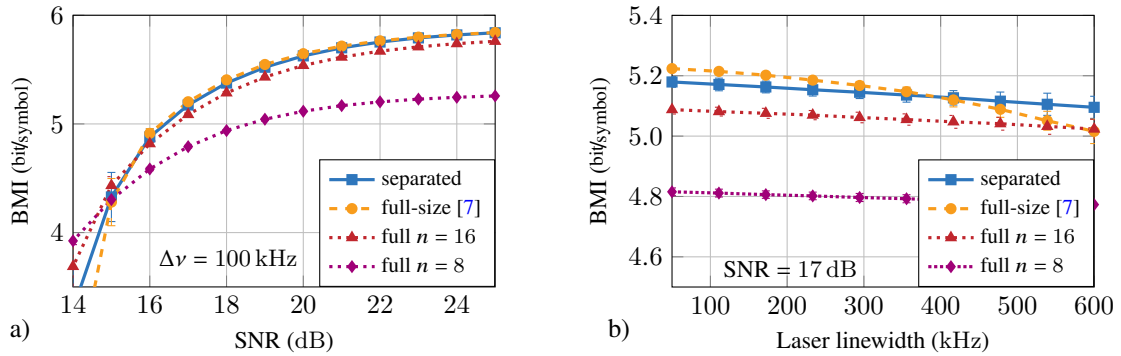

Fig. 3: Performance metrics for $M = 64$ with the separated, full-sized, and small demappers

demapper uses only 1 hidden layer and 8 nodes for Im-NN and Re-NN. We explain this matched performance with the fact that the transmit constellation, which we obtained with the separated demapper, forms a real Gray mapping, while the one obtained with the fully-connected demappers only approximately obtain a Gray mapping. Especially when using a lower number of weights in the neural demapper, the performance of the fully-connected demapper drops significantly compared to the separated demapper.

5. Conclusion

By separating the neural demapper in in-phase and quadrature, we are able to apply GCS to communication systems impaired by Wiener phase noise and increase spectral efficiency and robustness of these constellations while keeping the implementation complexity of the neural demapper low. This work shows how communication systems that are implemented and optimized with neural networks can be simplified to obtain a lower complexity and how to implement them in a complexity-constrained environment.

References

1. B. Karanov, M. Chagnon, F. Thouin, T. A. Eriksson, H. Bülow, D. Lavery, P. Bayvel, and L. Schmalen, "End-to-end deep learning of optical fiber communications," *J. Light. Technol.* **36**, 4843–4855 (2018).
2. T. Uhlemann, S. Cammerer, A. Span, S. Doerner, and S. ten Brink, "Deep-learning autoencoder for coherent and nonlinear optical communication," in *21th ITG-Symposium Photonic Networks,* (2020).
3. H. Dzieciol, G. Liga, E. Sillekens, P. Bayvel, and D. Lavery, "Geometric shaping of 2-D constellations in the presence of laser phase noise," *J. Light. Technol.* **39**, 481–490 (2021).
4. O. Jovanovic, M. P. Yankov, F. Da Ros, and D. Zibar, "End-to-end learning of a constellation shape robust to channel condition uncertainties," *J. Light. Technol.* **40**, 3316–3324 (2022).
5. M. P. Yankov, O. Jovanovic, and D. Zibar, "Recent advances in constellation optimization for fiber-optic channels," in *European Conference on Optical Communication (ECOC),* (2022).
6. T. Pfau, S. Hoffmann, and R. Noe, "Hardware-efficient coherent digital receiver concept with feedforward carrier recovery for M-QAM constellations," *J. Light. Technol.* **27**, 989–999 (2009).
7. A. Rode, B. Geiger, and L. Schmalen, "Geometric constellation shaping for phase-noise channels using a differentiable blind phase search," in *Optical Fiber Communications Conference and Exhibition (OFC),* (2022).
8. O. Jovanovic, M. P. Yankov, F. Da Ros, and D. Zibar, "End-to-end learning of a constellation shape robust to variations in SNR and laser linewidth," in *European Conference on Optical Communication (ECOC),* (2021).
9. A. Rode and L. Schmalen, "Optimization of geometric constellation shaping for Wiener phase noise channels with varying channel parameters," in *European Conference on Optical Communication (ECOC),* (2022).
10. M. Schaedler, S. Calabrò, F. Pittalà, C. Bluemm, M. Kuschnerov, and S. Pachnicke, "Neural network-based soft-demapping for nonlinear channels," in *Optical Fiber Communications Conference and Exhibition (OFC),* (2020).
11. S. Cammerer, F. Ait Aoudia, S. Dörner, M. Stark, J. Hoydis, and S. ten Brink, "Trainable communication systems: concepts and prototype," *IEEE Trans. Commun.* **68**, 5489–5503 (2020).