CHALMERS
UNIVERSITY OF TECHNOLOGY

Model-Based End-to-End Learning for WDM Systems With Transceiver Hardware Impairments

Citation for the original published paper (version of record):
Song, J., Häger, C., Schröder, J. et al (2022). Model-Based End-to-End Learning for WDM Systems With Transceiver Hardware Impairments. IEEE Journal of Selected Topics in Quantum Electronics, 28(4). http://dx.doi.org/10.1109/JSTQE.2022.3163474

Model-Based End-to-End Learning for WDM Systems With Transceiver Hardware Impairments

Jinxiang Song, *Student Member, IEEE*, Christian Häger, *Member, IEEE*, Jochen Schröder, *Member, IEEE*, Alexandre Graell i Amat, *Senior Member, IEEE*, and Henk Wymeersch, *Senior Member, IEEE*

(Invited Paper)

***Abstract*—We propose an autoencoder (AE)-based transceiver for a wavelength division multiplexing (WDM) system impaired by hardware imperfections. We design our AE following the architecture of conventional communication systems. This enables to initialize the AE-based transceiver to have similar performance to its conventional counterpart prior to training and improves the training convergence rate. We first train the AE in a single-channel system, and show that it achieves performance improvements by putting energy outside the desired bandwidth, and therefore cannot be used for a WDM system. We then train the AE in a WDM setup. Simulation results show that the proposed AE significantly outperforms the conventional approach. More specifically, it increases the spectral efficiency of the considered system by reducing the guard band by 37% and 50% for a root-raised-cosine filter-based matched filter with 10% and 1% roll-off, respectively. An ablation study indicates that the performance gain can be ascribed to the optimization of the symbol mapper, the pulse-shaping filter, and the symbol demapper. Finally, we use reinforcement learning to learn the pulse-shaping filter assuming that the channel model is unknown. Simulation results show that the reinforcement-learning-based algorithm achieves similar performance as the standard supervised end-to-end learning approach assuming perfect channel knowledge.**

***Index Terms*—Autoencoders, deep learning, digital signal processing, end-to-end learning, reinforcement learning, wavelength-division multiplexing.**

I. Introduction

The ever-growing demand for higher data rates drives the rapid development of optical fiber communication systems. One of the most important developments is wavelength division multiplexing (WDM) transmission, where parallel data channels are transmitted on different wavelengths simultaneously. The throughput of modern WDM systems often exceeds tens of $\mathrm{Tb/s}$ with more than 100 channels [2]. However, the overall bandwidth of fiber systems is limited by the bandwidth of erbium-doped fiber amplifiers (EDFAs) that periodically amplify the signals along the fiber link [3]. Optimizing the spectral efficiency (SE), i.e., the number of bits that can be transmitted per unit time and frequency, is therefore crucial to further increase the throughput of fiber optical systems.

Over the last decade, most works have focused on increasing the per-channel SE via advanced modulation formats using coherent detection. The fiber nonlinearity and hardware impairments, such as the effective number of bits (ENOBs) of the digital-to-analog converter (DAC), however, severely limit the per-channel SE. Furthermore, spectrum gaps between individual channels, which are often referred to as guard bands, waste significant bandwidth and limit the overall system throughput. Hence, the guard bands between channels need to be minimized. The most promising solution has been the application of flexible grids, which allows for transmission with flexible channel bandwidths thus enabling simultaneous transmission of mixed bit rates [4] and allowing to reduce SE loss from guard bands for optical filtering.

To minimize the guard bands between channels, it is common to employ pulse shaping to create a near-rectangular spectrum in the frequency domain with a bandwidth close to the symbol rate. However, in practice, generating a rectangular spectrum is difficult due to the finite pulse-shaping (PS) filter and transceiver hardware impairments, requiring computation expensive digital signal processing to eliminate performance degradation caused by inter-channel interference (ICI) [5]. Guard bands therefore remain a major contributor to SE loss in WDM systems.

In recent years, the rapid improvement of machine learning techniques has led to a resurgence of interest in applying deep learning techniques for communication systems [6], [7]. Most work has focused on supervised learning for a *specific functional block*, e.g., modulation recognition [8], carrier recovery [9], and fiber nonlinearity mitigation [10], with the aim of finding better performing (or less complex) algorithms by replacing the conventional model-based methods with neural networks (NNs). In contrast to focusing on specific functional blocks, *end-to-end learning* has been proposed to design the transmitter and receiver jointly [11]. The key idea is to interpret the transceiver design as a reconstruction task, whereby the transmitter and the receiver can be implemented as an autoencoder (AE) and thus jointly optimized in a data-driven manner without the need for prior mathematical modeling and analysis [12]. This method has led to several applications for both wireless [11]–[13] and optical communications [14]–[16]. A broad, but non-exhaustive overview of existing work is listed in Table I. We observe that (i) a majority of works relate to wireless rather than optical communication; (ii) geometric

Parts of this paper have been presented at the *Optical Fiber Communication Conference and Exhibition (OFC)*, San Diego, California, USA, 2021 [1]

This work was supported by the Knut and Alice Wallenberg Foundation, grant No. 2018.0090, and the Swedish Research Council under grant No. 2018-0370. *(Corresponding author: Jinxiang Song)*

Jinxiang Song, Christian Häger, Alexandre Graell i Amat, and Henk Wymeersch are with the Department of Electrical Engineering, Chalmers University of Technology, 41296 Gothenburg, Sweden (emails: {jinxiang, christian.haeger, alexandre.graell, henkw}@chalmers.se).

Jochen Schröder is with the Department of Microtechnology and Nanoscience, Chalmers University of Technology, 41296 Gothenburg, Sweden (email: jochen.schroeder@chalmers.se)

TABLE I: Applications of end-to-end AE-learning in communication systems

	Ref.	year	Application	isolated ch.	ICI ch.	sim.	exp.	Description
Wireless	[11]	2017	geom. shaping	✓		✓		const. mapper/demapper training over an AWGN channel
	[12]	2017	geom. shaping	✓		✓	✓	mapper/demapper training in sim., demapper tuning in exp.
	[20]	2017	geom. shaping	✓		✓		const. mapper/demapper training for PAPR reduction
	[21]	2017	geom. shaping & precoding		✓	✓		MIMO precoding/decoding
	[13]	2018	geom. shaping	✓		✓		mapper/demapper training for OFDM system
	[22]	2018	geom. shaping	✓		✓		mapper/demapper training over a learned channel via GAN
	[23]	2018	geom. shaping	✓		✓	✓	mapper/demapper training without knowing the channel model
	[24]	2019	geom. shaping	✓		✓		mapper/demapper training for PAPR reduction
	[25]	2019	joint channel/source coding	✓		✓		Joint channel and source coding/decoding
	[26]	2019	geom. & prob. shaping	✓		✓		Joint geom. and prob. shaping/demapping
	[27]	2020	geom. shaping & coding	✓		✓	✓	mapper/demapper learning and error correction code design
	[28]	2020	geom. shaping	✓		✓		mapper/demapper training for OFDM and multi-user system
	[19]	2021	geom. shaping & waveform	✓	✓	✓		Joint transceiver training
	[29]	2021	geom. shaping	✓		✓		mapper/demapper training for OFDM system
Fiber optic	[14]	2018	geom. shaping	✓		✓		mapper/demapper training for the nonlinear fiber channel
	[15]	2018	geom. shaping	✓		✓		mapper/demapper training for the fiber channel
	[17]	2018	geom. shaping & waveform	✓		✓	✓	Joint tranceiver learning for IM/DD system
	[30]	2019	geom. shaping	✓		✓		mapper/demapper training for optimizing GMI
	[31]	2019	geom. shaping & waveform	✓		✓	✓	Joint transceiver learning for IM/DD system
	[18]	2020	geom. shaping & waveform	✓		✓		Joint transceiver learning for single channel transmission
	[32]	2020	geom. shaping	✓		✓		mapper/demapper training for optimizing GMI
	[33]	2020	Prob. shaping	✓		✓		Prob. shaping
	[1]	2021	geom. shaping & waveform	✓	✓	✓		Joint transceiver design for superchannel systems
	[16]	2021	geom. shaping	✓		✓		mapper/demapper training for varying SNR and laser linewidth
	This work	2021	geom. shaping & waveform	✓	✓	✓		Joint transceiver design for densely-spaced WDM systems

isolated ch.: the channel does not suffer from ICI; ICI ch.: channel that suffers from ICI; sim.: simulation; exp.: experiment. References are ordered by year of publication.

constellation shaping for different channels and applications has been the main focus; (iii) there are few experimental validations. Studies that also learn waveforms and equalizers are limited to [17]–[19]. In [17], the whole transceiver is implemented as an AE, and transmission is demonstrated over a short-haul intensity modulation/direct detection (IM/DD) system. However, the NN in [17] is used as a "black-box" and it is difficult to interpret the learned solution. In [18], the transmitter is implemented as a trainable constellation mapper combined with a trainable PS filter, and it is shown that the PS filter can be learned to mitigate chromatic dispersion and Kerr effect. An explicit low-pass filter is used to reduce information loss and thus avoid out-of-band (OOB) emissions. A related approach has recently been applied in [19], where flexible constellations and waveforms for wireless dispersive channels under OOB power leakage constraints were learned.

In this paper, we apply end-to-end learning to a multi-channel WDM system. Similar to [18], [19] (for a single-channel system), we consider designing several transceiver blocks —constellation mapper, PS filter, digital pre-distortion (DPD), and demapper—jointly. Such an AE design incorporates the expert domain knowledge of conventional communication systems and therefore allows for (i) training speed improvements via meaningful parameter initialization and (ii) performance gain explanation through an ablation study. The main contributions of this paper are:

- We propose a novel end-to-end AE for WDM transceivers with non-ideal DAC and in-phase and quadrature modulator (IQM). We decompose the transmitter NN into a concatenation of small (simple) NNs, each corresponding to a functional block of a conventional communication system. Our approach differs from [18], [19] in terms of the considered hardware impairments and how OOB emissions are accounted for: instead of a low-pass filter [18] or a constraint [19], we show that the AE automatically learns to avoid/adapt OOB emissions to minimize the end-to-end loss.
- We highlight the potential pitfalls when using end-to-end AE-learning for designing hardware-impaired communication systems. In particular, we show that when the AE is trained for a single-channel system, it achieves performance improvements by putting energy outside the desired signal bandwidth, which would cause large ICI in WDM systems when the channels are closely spaced. We demonstrate that if the AE is instead trained with three channels, it learns to limit ICI while still outperforming the considered baseline. However, care must be taken for the sampling rates or bandwidths used during training to match experimental constraints to avoid unrealistic gains.
- We conduct a thorough ablation study and show that the performance improvement of the AE-based system is ascribed to the optimization of the constellation mapper, the PS filter, and the demapper. Therefore, we show that our proposed method increases the interpretability compared to conventional AE-based systems. Additionally, we provide reproducible open-source implementations of our AEs and benchmark scheme.[1]
- We extend the model-free training algorithm proposed in [23], [34], so that the reinforcement learning (RL) based transmitter training algorithm can be applied to train the PS filter, for which memory effects need to be considered. The resulting training algorithm is shown to achieve similar performance to the standard end-to-end learning approach assuming a perfect channel model. This opens the door toward experimental implementation of the proposed AE.

The remainder of this paper is structured as follows. In Section II, we give a brief introduction to DL basics and the

[1]The complete source code to reproduce all results in this paper is available at https://github.com/JSChalmers/AE-Based-WDM-Transceivers.

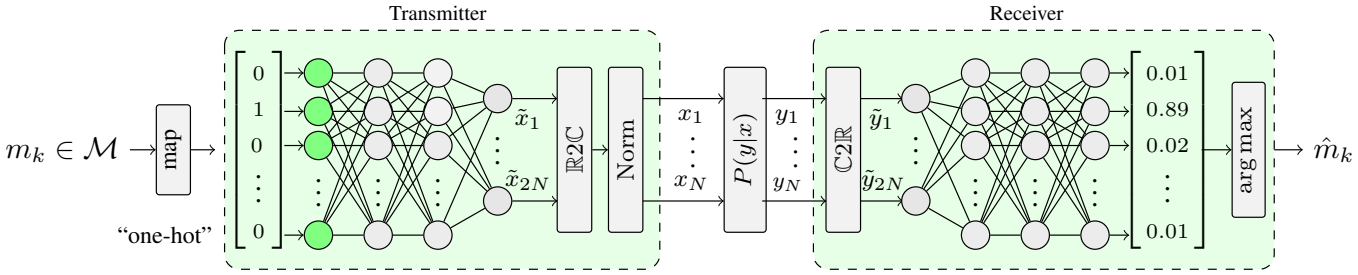

Fig. 1: Example of an AE-based communication system, where the transmitter and receiver are implemented by a pair of fully connected NNs. $\mathbb{R}2\mathbb{C}$: real-to-complex; $\mathbb{C}2\mathbb{R}$: complex-to-real; Norm: normalization.

concept of AE-based communications. Then, in Section III, we introduce the generic setup of closely-spaced WDM systems and the main hardware limitations. Section IV introduces the proposed AE-based WDM system and simulation results are provided in Section V. Finally, the paper is concluded in Section VI.

Notation: $\mathbb{Z}$, $\mathbb{R}$, and $\mathbb{C}$ denote the sets of integers, real numbers, and complex numbers, respectively. Column vectors will be denoted with lower case letters in bold (e.g., $\boldsymbol{x}$), with x_n referring to the n-th entry in $\boldsymbol{x}$; $\|\boldsymbol{x}\|_2^2$ denotes the Euclidean norm of $\boldsymbol{x}$ and $\boldsymbol{x}_n^{(L)}$ denotes the column vector consisting of $(n-L)$-th to $(n+L)$-th elements of $\boldsymbol{x}$; $|\cdot|$ returns the absolute value of a real number, and $|\Im\{\boldsymbol{x}\}|$ and $|\Im\{\boldsymbol{x}\}|$ return the absolute value of the real and imaginary part of each element in $\boldsymbol{x}$, respectively; $(\cdot)^\top$ and $(\cdot)^\mathsf{H}$ denote transpose and conjugate transpose, respectively. Matrices will be denoted in bold capitals (e.g., $\boldsymbol{X}$), and $\boldsymbol{I}_N$ denotes identity matrix of size N; $[a,b]^M$ is the M-fold Cartesian product of the interval $[a,b]$. Sets will be denoted in calligraphic capitals (e.g., $\mathcal{X}$), with $|\mathcal{X}|$ referring to the cardinally of $\mathcal{X}$; $\mathcal{X}\times\mathcal{Y}$ denotes the Cartesian product of $\mathcal{X}$ and $\mathcal{Y}$. Lastly, $\mathbb{E}\{\cdot\}$ denotes the expectation operator.

II. Deep Learning and Autoencoder-Based Communication systems

In this section, we start by reviewing the general theory behind DL, followed by a brief introduction to the concept of AE-based communication systems. Then, we introduce the training of AE-based communication systems under two assumptions: (i) the channel model is known and differentiable, and (ii) the channel model is unknown or not differentiable.

A. Neural Networks and Gradient-Based Learning

1) Feedforward NN: A feedforward NN with K layers is a parametric function $f(\boldsymbol{r}_0;\boldsymbol{\theta}):\mathbb{R}^{N_0}\to\mathbb{R}^{N_K}$ that maps an input vector $\boldsymbol{r}_0\in\mathbb{R}^{N_0}$ to an output vector $\boldsymbol{r}_K\in\mathbb{R}^{N_K}$ through K sequential processing steps according to

$$\boldsymbol{r}_k = f_k(\boldsymbol{r}_{k-1};\boldsymbol{\theta}_k),\quad k=\{1,\dots,K\}, \tag{1}$$

where $f_k(\boldsymbol{r}_{k-1};\boldsymbol{\theta}_k):\mathbb{R}^{N_{k-1}}\to\mathbb{R}^{N_k}$ is the mapping carried out by the k-th layer. Here, the mapping of the k–th layer is defined by the set of parameters $\boldsymbol{\theta}_k$, and the entire NN is defined by $\boldsymbol{\theta}=\{\boldsymbol{\theta}_1,\dots,\boldsymbol{\theta}_K\}$. A commonly used type of feedforward NN is the fully connected NN in which all layers have the form

$$f_k(\boldsymbol{r}_{k-1};\boldsymbol{\theta}_k) = \sigma(\boldsymbol{W}_k\boldsymbol{r}_{k-1}+\boldsymbol{b}_k), \tag{2}$$

where $\boldsymbol{W}_k\in\mathbb{R}^{N_k\times N_{k-1}}$ is a weight matrix, $\boldsymbol{b}_k\in\mathbb{R}^{N_k}$ is a bias vector, and $\sigma(\cdot)$ is a point-wise *activation* function. Hence, the set of trainable parameters of the k-th layer is $\boldsymbol{\theta}_k=\{\boldsymbol{W}_k,\boldsymbol{b}_k\}$. An example of a fully connected NN is shown in the transmitter and the receiver in Fig. 1.

2) Gradient-based learning: Training of the NN can be performed in an iterative fashion with data-driven gradient-based optimization methods. Given a set of labeled training data $\mathcal{D}\subset\{\mathcal{X}\times\mathcal{Y}\}$, where $\mathcal{X}$ and $\mathcal{Y}$ are the input and output alphabets, the training objective is to find the set of parameters $\boldsymbol{\theta}$ such that the average loss

$$\mathcal{L}_\mathcal{D}(\boldsymbol{\theta}) = \frac{1}{|\mathcal{D}|}\sum_{(x,y)\in\mathcal{D}}\ell(f(x;\boldsymbol{\theta}),y) \tag{3}$$

between the NN output $\hat{y}=f(x,\boldsymbol{\theta})$ and the true label $y\in\mathcal{Y}$ is minimized. Here, $|\mathcal{D}|$ is the size of the training data set and $\ell(f(x;\boldsymbol{\theta}),y)$ is the per-example loss function associated with returning the output $\hat{y}=f(x;\boldsymbol{\theta})$ when y is the true label. In practice, when the training data set $\mathcal{D}$ is large, computing the gradients of the average loss over the whole training data set is computationally expensive, and the parameter set $\boldsymbol{\theta}$ is commonly optimized by using stochastic gradient descent (SGD) or its variants as follows. For each training iteration t, a minibatch $\mathcal{B}_t\subset\mathcal{D}$ is sampled from $\mathcal{D}$. Then, the parameter set $\boldsymbol{\theta}$ is updated according to

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \alpha\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}_t), \tag{4}$$

where $\alpha>0$ is the learning rate. In practice, SGD sometimes suffers from slow convergence rate due to problems like small gradients at suboptimal values of $\boldsymbol{\theta}$. To improve the convergence rate of SGD, many variants of SGD using momentum [35] or adaptive learning rate [36] have been proposed.

B. End-to-End AE Learning-Based Communication Systems

1) AE-based Communication Systems: End-to-end learning of AE-based communication systems was originally proposed in [11], where the transceiver for a given channel with channel law $p(\boldsymbol{y}|\boldsymbol{x})$ is implemented by a pair of NNs $f_{\boldsymbol{\tau}}:\mathcal{M}\to$

$\mathbb{C}^N$ and $f_{\boldsymbol{\rho}} : \mathbb{C}^N \to [0,1]^M$. Here, $\mathcal{M} = \{1, \ldots, M\}$ is the message set, N is the number of complex channel uses, and $\boldsymbol{\tau}$ and $\boldsymbol{\rho}$ are the sets of trainable NN parameters. Fig. 1 depicts the general setup of an AE-based communication system,

Transmitter: Given a message $m_k \in \mathcal{M}$, it is first encoded as an M-dimensional "one-hot" vector, where the m_k-th element is 1 and all the others are 0.[2] Then, the transmitter NN takes this "one-hot" vector as input and generates a vector of $2N$ outputs $\tilde{\boldsymbol{x}}_k = f_{\boldsymbol{\tau}}(m_k) \in \mathbb{R}^{2N}$, where the $2N$ outputs correspond to the real and imaginary part of the transmitted vector $\boldsymbol{x}_k \in \mathbb{C}^N$, which is obtain by a real-to-complex conversion layer [34]. The average transmit power constraint $\mathbb{E}\{||\boldsymbol{x}_k||^2\} \leq NP_T$, where P_T is the average transmit power per channel use, is enforced by a normalization layer [11].

Receiver: The symbol $\boldsymbol{x}_k$ is sent over the channel in N complex channel uses, after which $\boldsymbol{y}_k \in \mathbb{C}^N$ is observed at the receiver. The receiver first converts $\boldsymbol{y}_k$ to a real-valued vector $\tilde{\boldsymbol{y}}_k \in \mathbb{R}^{2N}$ using a complex-to-real conversion layer [34], after which an M-dimensional probability vector $\boldsymbol{q}_k \in [0,1]^M$ is generated according to $\boldsymbol{q}_k = f_{\boldsymbol{\rho}}(\boldsymbol{y}_k)$. Here, the components of $\boldsymbol{q}_k$ can be interpreted as the estimated posterior probabilities of the messages. Finally, the transmitter generates the estimate of the transmitted message according to $\hat{m}_k = \arg\max_m [\boldsymbol{q}_k]_m$, where $[\boldsymbol{x}]_m$ returns the m-th element of $\boldsymbol{x}$.

2) End-to-End Training With a Known Channel Model: To optimize the transmitter and receiver parameters, it is crucial to have a suitable optimization criterion. Due to the fact that the optimization relies on the empirical computation of gradients, a criterion like block error rate (BLER), i.e., $\Pr\{\hat{m}_k \neq m_k\}$, cannot be used directly (as the BLER is not differentiable). Instead, a commonly used criterion is the cross-entropy loss [11], defined by

$$\mathcal{L}(\boldsymbol{\tau}, \boldsymbol{\rho}) = -\mathbb{E}\{\log[f_{\boldsymbol{\rho}}(\boldsymbol{y})]_m\}, \tag{5}$$

where the dependence of $\mathcal{L}(\boldsymbol{\tau}, \boldsymbol{\rho})$ on $\boldsymbol{\tau}$ is implicit through the distribution of the channel output $\boldsymbol{y}_k$, which is a function of the channel input $f_{\boldsymbol{\tau}}(m_k)$. We remark that the cross-entropy loss can be used to lower bound the mutual information as follow [14], [15]:

$$\begin{aligned} I(M;Y) &= \mathbb{E}\{\log[\frac{p(\boldsymbol{y}|m)}{p(\boldsymbol{y})}]\} \qquad (6)\\ &= \mathbb{E}\{\log[\frac{p(\boldsymbol{y}|m)\,p(m)}{p(\boldsymbol{y})\,[f_{\boldsymbol{\rho}}(\boldsymbol{y})]_m} \cdot \frac{[f_{\boldsymbol{\rho}}(\boldsymbol{y})]_m}{p(m)}]\} \\ &= \mathrm{KL}(p(m,\boldsymbol{y})\|p(\boldsymbol{y})\,[f_{\boldsymbol{\rho}}(\boldsymbol{y})]_m) + \mathbb{E}\{\log[\frac{[f_{\boldsymbol{\rho}}(\boldsymbol{y})]_m}{p(m)}]\} \\ &\geq \mathbb{E}\{\log[f_{\boldsymbol{\rho}}(\boldsymbol{y})]_m - \log[p(m)]\} \\ &= H(M) - \mathcal{L}(\boldsymbol{\tau}, \boldsymbol{\rho})\,, \end{aligned}$$

where the inequality comes from the fact that the Kull-Leibler divergence is non-negative, $\mathrm{KL}(p(m,\boldsymbol{y})\|p(\boldsymbol{y})\,[f_{\boldsymbol{\rho}}(\boldsymbol{y})]_m) \geq 0$. And the entropy $H(M)$ is a constant assuming the transmitted messages are uniformly distributed. Therefore, minimizing the cross-entropy loss is equivalent to maximizing the lower bound of the mutual information.

The transmitter and receiver parameters are optimized in an iterative fashion as follows. In each training iteration t, the transmitter maps a minibatch of $|\mathcal{B}_t|$ randomly chosen uniformly distributed training examples to symbols and then sends them over the channel. The receiver takes the channel observations $\boldsymbol{y}_1, \cdots, \boldsymbol{y}_{|\mathcal{B}_t|}$ as input and generates $|\mathcal{B}_t|$ probability vectors $f_{\boldsymbol{\rho}}(\boldsymbol{y}_1), \cdots, f_{\boldsymbol{\rho}}(\boldsymbol{y}_{|\mathcal{B}_t|})$. Finally, the receiver computes the empirical cross-entropy loss associated with the $|\mathcal{B}_t|$ training examples according to

$$\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\tau}, \boldsymbol{\rho}) = -\frac{1}{|\mathcal{B}_t|} \sum_{k=1}^{|\mathcal{B}_t|} \log[f_{\boldsymbol{\rho}}(\boldsymbol{y}_k)]_{m_k}, \tag{7}$$

and the transmitter and receiver parameters are optimized following (4). This training process is repeated iteratively until a certain criterion is satisfied (e.g., a fixed number of training iterations, or a fixed number of iterations during which the loss has not significantly decreased).

3) Training Without a Channel Model: In case the channel is unknown or not differentiable, e.g., an experimental channel, the transmitter optimization becomes challenging due to the fact that the gradient of the instantaneous channel transfer function is unknown, thus hindering the numerical computation of the transmitter gradients. One way to circumvent this limitation is to first learn a surrogate channel model, e.g., through supervised learning [39], [40] or an adversarial process [22], [41], and use the surrogate model to train the transmitter. However, the performance of the resulting system severely degrades if the surrogate model deviates from the real channel. A different approach based on a stochastic transmitter was proposed in [23], [34]. For this approach, the transmitter is regarded as an RL agent that learns to communicate with the receiver by taking random actions (i.e., waveforms) to interact with the environment (i.e., the communication channel and the receiver). The transmitter learning objective is to find the optimal actions that minimize some cost (e.g., the cross-entropy loss). The transmitter and receiver training are performed in an alternating fashion which we review next.

Receiver training: The receiver training is similar as before. However, this time, the transmitter parameters $\boldsymbol{\tau}$ are assumed to be fixed. At each training iteration, the transmitter maps a minibatch of $|\mathcal{B}_t|$ uniformly distributed training examples to symbols and sends them over the channel. The receiver takes the channel observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{|\mathcal{B}_t|}$ as input and generates $|\mathcal{B}_t|$ probability vectors $f_{\boldsymbol{\rho}}(\boldsymbol{y}_1), \ldots, f_{\rho}(\boldsymbol{y}_{|\mathcal{B}_t|})$. Then, the receiver takes one optimization step according to $\boldsymbol{\rho}_{t+1} = \boldsymbol{\rho}_t - \alpha \nabla_{\boldsymbol{\rho}} \mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\tau}_t, \boldsymbol{\rho}_t)$, where $\boldsymbol{\tau}_t$ is fixed during receiver training. This training process is repeated iteratively until a certain stop criterion is satisfied.

Transmitter training: For the transmitter optimization, the receiver parameters are assumed to be fixed. At each training iteration, the transmitter performs the symbol mapping as before. In order to allow for the transmitter gradients computation, a small Gaussian perturbation is applied such that $\tilde{\boldsymbol{x}} = \boldsymbol{x} + \boldsymbol{w}$, $\boldsymbol{w} \sim \mathcal{CN}(0, \sigma_p^2 \boldsymbol{I}_N)$, is sent over the channel.

[2]The "one-hot" encoding is the standard way of representing categorical values in most machine learning algorithms [37] and facilitates the minimization of the symbol error rate (SER). However, the dimension of the "one-hot" vector grows exponentially with the number of bits in each message and therefore increases the NN size. Alternative embeddings [38] and multi-hot sparse categorical cross-entropy loss can be used to alleviate this problem.

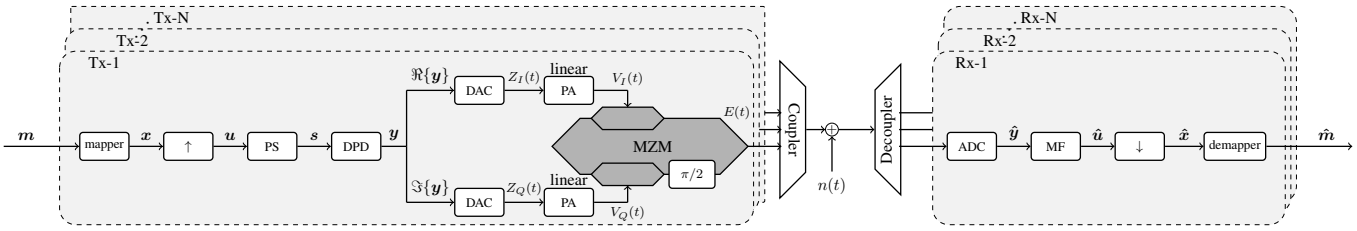

Fig. 2: Block diagram showing the conventional WDM system (↑: upsampling, ↓: downsampling). The mapper, DPD, and demapper operate on each individual entries of the input sequence, while the PS filter operates on a sequence of $2L_1+1$ signals, where $2L_1+1$ is the PS filter taps. Note, to allow close channel spacing close to the symbol rate, we assume that there are no optical filters or multiplexers.

Therefore, the transmitter can be interpreted as stochastic and is described by

$$\pi_{\boldsymbol{\tau}}(\tilde{\boldsymbol{x}}_k|m_k) = \frac{1}{(\pi\sigma_p^2)^N}\exp\left(-\frac{||\tilde{\boldsymbol{x}}_k - f_{\boldsymbol{\tau}}(m_k)||_2^2}{\sigma_p^2}\right). \quad (8)$$

Based on the received channel observations, the receiver computes per-example losses $\ell_k = -\log([f_{\boldsymbol{\rho}}(\boldsymbol{y}_k)]_{m_k})$, and sends them back to the transmitter. Finally, the transmitter parameters $\boldsymbol{\tau}$ are updated according to $\boldsymbol{\tau}_{t+1} = \boldsymbol{\tau}_t - \alpha\nabla_{\boldsymbol{\tau}}\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\tau}_t)$, where $\nabla_{\boldsymbol{\tau}}\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\tau})$ is approximated by

$$\nabla_{\boldsymbol{\tau}}\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\tau}) = \frac{1}{N_T}\sum_{k=1}^{N_T}\ell_k\nabla_{\boldsymbol{\tau}}\log\pi_{\boldsymbol{\tau}}(\tilde{\boldsymbol{x}}_k|m_k), \quad (9)$$

for which a theoretical justification can be found in [34]. Similar to the receiver training, the transmitter learning process is repeated iteratively until a certain stopping criterion is satisfied. Then, the alternating optimization continues again with the receiver learning.

III. WDM System and Main Hardware Limitations

A. System Model

Fig. 2 illustrates the considered WDM system, where parallel data streams are transmitted simultaneously by modulating them onto different wavelengths. Here, the different wavelength channels are assumed to share the same hardware configurations. For each channel, a sequence of $|\mathcal{B}_t|$ messages $\boldsymbol{m} \in \mathcal{M}^{|\mathcal{B}_t|}$, where $\mathcal{M} = \{1,\dots,M\}$, are mapped individually to constellation points according to a constellation $\mathcal{C} \in \mathbb{C}^M$, to form the sequence of baseband symbols $\boldsymbol{x} \in \mathbb{C}^{|\mathcal{B}_t|}$. The baseband symbols $\boldsymbol{x}$ are then upsampled to get $\boldsymbol{u} \in \mathbb{C}^{|\mathcal{B}_t|R}$, after which a PS filter is applied to get the discrete-time baseband signals $\boldsymbol{s} \in \mathbb{C}^{|\mathcal{B}_t|R}$.[3] Here, R is the upsampling rate, and the R–times upsampling is performed by inserting $R-1$ zeros between every two adjacent symbols. To mitigate the performance degradation caused by the hardware imperfections (in this paper ENOB of the DAC and IQM nonlinearity), a DPD algorithm is applied. Then, the real and imaginary part of the pre-distorted signals $\boldsymbol{y} \in \mathbb{C}^{|\mathcal{B}_t|R}$ are separately fed to the DACs of the in-phase and quadrature branches. The DACs outputs $Z_I(t)$ and $Z_Q(t)$ are then separately amplified to drive the IQM, where the driving voltages of the in-phase and quadrature branches are denoted by $V_I(t)$ and $V_Q(t)$, respectively. Finally, the output fields $E(t)$ of the Mach-Zehnder modulators (MZMs) from different channels are combined to generate the WDM signals.[4] Similar to [42]–[44], the channel model we consider in this paper is restricted to a back-to-back setup, and only additive white Gaussian noise (AWGN) with constant power is added to simulate the noise introduced by the booster amplifier. At the receiver, the received signals are first filtered by a low-pass filter (to avoid frequency aliasing that may occur during the sampling) and then sampled by an analog-to-digital converter (ADC) with rate R. Then, the digitized channel observations $\hat{\boldsymbol{y}} \in \mathbb{C}^{|\mathcal{B}_t|R}$ are convolved with a matched filter (MF) and then down-converted with a factor R to have one sample per symbol. Finally, the downsampled signals $\hat{\boldsymbol{x}} \in \mathbb{C}^{|\mathcal{B}_t|}$ are individually mapped to the estimates $\hat{\boldsymbol{m}} \in \mathcal{M}^{|\mathcal{B}_t|}$ of the transmitted messages. Note that, as optical filters and multiplexers would prevent close channel spacing due to their finite response, we assume that channels are combined using broadband passive couplers. Thus, there are no optical filters in our system; such a system is often referred to as *superchannel* system.

IQM Model: The coherent optical transmitter used for high-order modulation schemes such as M–QAM, M–PAM is often based on a dual parallel MZM. For an ideal dual parallel MZM biased at the null point, it has been shown that its transfer function becomes [45]

$$E(t) = E_0\left[\sin\left(\frac{\pi V_I(t)}{2V_\pi}\right) + j\sin\left(\frac{\pi V_Q(t)}{2V_\pi}\right)\right], \quad (10)$$

where E_0 is the amplitude of the magnitude of the optical field, V_π is the required voltage difference to switch ON/OFF the modulator, and $V_I(t)$ and $V_Q(t)$ are the driving voltage of the in-phase and quadrature branches, respectively. The intrinsic sinusoidal form of the MZM leads to strong signal distortions when driving with a high peak voltage V_p, which must be compensated, e.g., by pre-distortion with an *arcsin* function. Alternatively, one can use a low-driving voltage to operate in the near-linear regime of the modulator. However, this significantly increases the modulator loss, which results in a degraded optical signal-to-noise ratio (SNR) after adding the booster amplifier noise.

[3]To guarantee the pulse-shaped signals to have the same length as the upsampled signals, $(N-1)/2$ samples are removed from both sides of the pulse-shaped signals, where N is the PS filter length. The same applies to the MF operation.

[4]To allow for placing the wavelength channels in the desired spectrum, additional resampling is required. The oversampling in this paper is implemented by zero-padding in the frequency domain to avoid frequency aliasing.

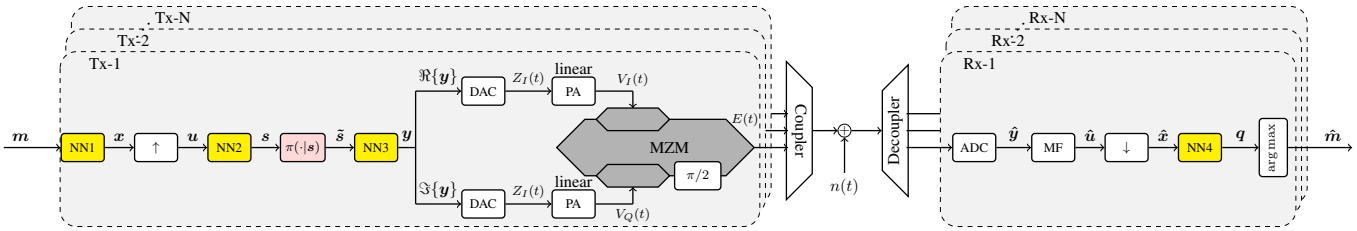

Fig. 3: Block diagram showing the end-to-end AE-learning based WDM system (↑: upsampling, ↓: downsampling). The trainable components are highlighted in yellow. NN1, NN3, and NN4 operate separately on each entry of the input sequence, while NN2 takes a vector of length $2L_1+1$ signals as input, where $2L_1+1$ is the PS filter length.

PA Model: The power amplifier (PA) used for amplifying the DAC outputs behaves as a nonlinear memory system, i.e., the PA output at any time instant depends on the current instantaneous input as well as the inputs at previous time instances. In coherent optical communication systems, separate PAs are used for the in-phase and quadrature branches. Here, without loss of generality, only the transfer function of the in-phase branch PA is denoted, while it should be noted that the quadrature branch PA will have similar form. Denoting the memory depth by L, the in-phase branch PA denoted by $f_{\text{PA}}:\mathbb{R}^{L+1}\to\mathbb{R}$, can be defined by

$$V_I(t)=f_{\text{PA}}(Z_I(t),\ldots,Z_I(t-L)), \tag{11}$$

where f_{PA} is a nonlinear function and $Z_I(t)$ is the DAC output of the in-phase branch. For an ideal PA without memory effect, its transfer function becomes $V_I(t)=GZ_I(t)$, where G is the PA gain.

DAC Model: DACs used for high-bandwidth optical communications typically have low resolution. Currently, devices on the market provide 8 nominal bits. However, due to the sampling and jitter effects, the noise introduced by quantization is usually enhanced. One parameter to assess the amount of noise introduced by the DAC is the ENOB, which is defined as [46]

$$\text{ENOB}=\frac{\text{SNDR}(\text{dB})-1.76}{6.02}, \tag{12}$$

where the signal-to-noise-plus-distortion ratio (SNDR) is a measurable quantity, and is typically around 35 dB. Typically, high-speed DACs with 8-bit nominal resolution can be translated into ENOB ≤ 6 for operation within the device bandwidth. However, it should be noted that ENOB is a varying quantity and it changes over frequency. In this paper, for the sake of simplicity, the ENOB is assumed to be constant over the considered bandwidth and is set to 6.[5] We model the ENOB noise introduced by the DAC as AWGN with variance determined by the ENOB of the device [47]

$$\sigma_q^2=\frac{1}{12}\left(\frac{E_{\text{peak}}}{2^{\text{ENOB}-1}-1}\right)^2, \tag{13}$$

where $E_{\text{peak}}=\max(\max(|\Re\{\boldsymbol{y}\}|),\max(|\Im\{\boldsymbol{y}\}|))$ is the peak amplitude of the input signals. Note that the finite bit-resolution of the DAC limits the strength of the *arcsin*-based pre-distortion that can be applied, because it increases the peak amplitude, thereby resulting in higher noise. Therefore, there exists an optimum DAC driving voltage which balances SNR degradation from MZM losses when driving in the linear regime of the modulator and SNR degradation from limited compensation of MZM nonlinearity when driving at high voltages.

[5]This is a reasonable assumption for current generation transceivers.

IV. Proposed End-to-End WDM System

In this section, we start by introducing the proposed AE implementation for the WDM system. We note that all our NNs operates on real numbers, while to cope with the fact that communication systems transmit complex baseband symbols, additional real-to-complex or complex-to-real conversion layers [34] are employed to perform conversion between real-valued and complex-valued vectors. The symbol rate and modulation formats are assumed to be the same for all channels, and we consider using the same AE configurations for all channels.

A. Autoencoder Design

In principle, the entire transmitter and receiver can be implemented as an AE and trained by end-to-end learning as proposed in [11]. However, this leads to:

(a) Difficulty in interpretation: In contrast to conventional communication systems, where the performance of each transmitter/receiver blocks can be measured separately, the AE implementation is a "black-box", and it is hard to interpret the learned solution and to quantify the origin of the performance improvement.

(b) High training complexity: The transmitter needs to perform several tasks, such as symbol mapping, PS, and pre-distortion jointly, and learning the transmitted waveform involves sequential input data, which significantly increases the NN size with the "one-hot" encoding being applied, therefore increasing the training complexity.

(c) Parameter initialization: It is difficult to know which parameter choice leads to good performance prior to training, and random parameters initialization can slow down or even completely stall the convergence process [48].

To address these issues, we design our AE following the architecture of conventional communication systems as shown in Fig. 3. The policy $\pi(\cdot|\boldsymbol{s})$ can be ignored for now. The transmitter NN is decomposed into a concatenation of three simpler (small) NNs, each corresponding to one functional

block of a conventional communication system. By doing this, the parameters of these NNs can be initialized such that they initially perform close to their conventional counterparts. Moreover, the "block-wise" transceiver NN design allows for an ablation study and therefore makes it possible to partially explain the learned solution. As a result, the proposed scheme has *decreased training complexity* and *increased interpretability* as compared to a conventional AE.[6]

1) Transmitter: At the transmitter, the symbol mapper, the PS filter, and the DPD of the conventional communication system are replaced by three NNs. We denote these three NNs by $f_{\boldsymbol{\theta}_1}(\cdot)$, $f_{\boldsymbol{\theta}_2}(\cdot)$, and $f_{\boldsymbol{\theta}_3}(\cdot)$, where $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2$, and $\boldsymbol{\theta}_3$ are the sets of trainable parameters. We define these three NNs in the following:

(i) NN1 $f_{\boldsymbol{\theta}_1}\colon \mathcal{M} \to \mathbb{C}$ maps each message $m_k \in \boldsymbol{m}$ to a constellation point according to $x_k = f_{\boldsymbol{\theta}_1}(m_k)$, where an average power constrain $\mathbb{E}\{|x_k|^2\} = 1$ is enforced.

(ii) NN2 $f_{\boldsymbol{\theta}_2}\colon \mathbb{C}^{2L_1+1} \to \mathbb{C}$ generates each of the pulse-shaped baseband signals according to $s_k = f_{\boldsymbol{\theta}_2}(\boldsymbol{u}_k^{(L_1)})$, where $\boldsymbol{u}_k^{(L_1)} = [u_{k-L_1}, \ldots, u_{k+L_1}]^\top$. Here, NN2 only has a single layer applying a linear activation function and can be interpreted as a standard finite impulse response (FIR) filter. Therefore, the generation of the pulse-shaped signal can be described by $s_k = \boldsymbol{\theta}_2^\top \boldsymbol{u}_k^{(L_1)}$.

(iii) NN3 $f_{\boldsymbol{\theta}_3}\colon \mathbb{C} \to \mathbb{C}$ generates each of the pre-distorted signals according to $y_k = f_{\boldsymbol{\theta}_3}(s'_k)$, where $f_{\boldsymbol{\theta}_3}(\cdot)$ operates separately on the in-phase and quadrature branches, and $-1 \leq \Re\{s'_k\}, \Im\{s'_k\} \leq 1$ is obtained by normalizing s_k according to $s'_k = s_k / \max\{\max\{|\Re\{\boldsymbol{s}\}|\}, \max|\Im\{\boldsymbol{s}\}|\}\}$, where $\boldsymbol{s}$ is the pulse-shaped signal sequence.

2) Receiver: At the receiver, only the symbol demapper is replaced by an NN, denoted by NN4 $f_{\boldsymbol{\theta}_4}\colon \mathbb{C} \to \mathcal{M}$, which maps each of the downsampled signal y_k to the estimate of the transmitted message as described in Section II-B1. We note that, in principle, the MF can also be implemented by an NN. In a real system, however, the MF is usually implemented as part of the adaptive equalizer, and we therefore have left it out of this discussion.

B. Learning With a Channel Model

Assuming that all transfer functions of the components in the considered system are known and differentiable, the system can be optimized via standard end-to-end AE-learning [11] by minimizing the Monte-Carlo approximation of the cross-entropy loss, defined by

$$\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3, \boldsymbol{\theta}_4) = \frac{1}{|\mathcal{B}_t|} \sum_{k=1}^{|\mathcal{B}_t|} \log[f_{\boldsymbol{\theta}_4}(y_k)]_{m_k}. \tag{14}$$

Similar to (5), the dependence of $\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3, \boldsymbol{\theta}_4)$ on $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3$ is implicit through the distribution of the downsampled signal y_k, which is a function of the channel input $g(\tilde{s}_k)$, where $g(\cdot)$ denotes the joint transfer function of the DAC, PA, and IQM, and $\tilde{s}_k$ is dependent on NNs 1–3 as can be seen in Fig. 3. For the optimization, in order to have a faster and more stable convergence, all NNs are first initialized to mimic their model-based counterparts via pre-training. Then, the sets of parameters $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3, \boldsymbol{\theta}_4$ are jointly optimized using the Adam optimizer [49].

[6]We note that such an AE implementation can potentially lead to performance degradation compared to the conventional AE, which we do not study in this paper.

Algorithm 1 Optimization of the pulse shaping filter

repeat
  ▷ Transmitter
  Symbol mapping and upsampling: $\boldsymbol{m} \to \boldsymbol{x} \to \boldsymbol{u}$
  Pulse shaping: $\boldsymbol{u} \to \boldsymbol{s}$
  Apply Gaussian: $\boldsymbol{s} \to \tilde{\boldsymbol{s}}$
  Apply DPD: $\tilde{\boldsymbol{s}} \to \boldsymbol{y}$
  Send $\boldsymbol{y}$
  ▷ Receiver
  Receive: $\hat{\boldsymbol{y}}$
  Matched filtering and downsampling: $\hat{\boldsymbol{y}} \to \hat{\boldsymbol{u}} \to \hat{\boldsymbol{x}}$
  Compute per example loss: ℓ_k
  Send ℓ_k
  ▷ Transmitter
  Receive ℓ_k
  Update NN2 parameters according to (17)
until Stop criterion is satisfied

C. Learning Without A Channel Model

In practice, training of the proposed AE in an experiment is challenging due to the fact that the instantaneous gradients of the physical channel are unknown. To solve this problem, we follow the alternative optimization approach that we reviewed in Section II-B3. The training of the demapper does not require differentiation of the channel can therefore be performed via supervised learning. For the transmitter training, since the transmitter consists of three NNs, one can perform the transmitter training by alternating between the optimization of the symbol mapper, the PS filter, and the DPD. In this paper however, we only focus on training of the PS filter, for which memory effects need to be taken into account. For the optimization of the mapper (i.e., NN1) or the DPD (i.e., NN3), we refer the reader to [23], [34] and our recent paper [50]. To that end, the parameters of the mapper, the DPD, and the demapper (i.e., NN4) are assumed to be pretrained and fixed during the PS filter training.

For the PS filter optimization, the training algorithm described in Section II-B3 cannot be used directly due to the memory introduced by the matched filtering. Therefore, we extend the training approach as follows. In each training iteration t, the transmitter generates a batch of $|\mathcal{B}_t|$ random uniformly distributed messages within one message vector $\boldsymbol{m} \in \mathcal{M}^{|\mathcal{B}_t|}$ and maps them individually to the baseband symbols after which R-time upsampling is applied. Then, the baseband transmitted signals are generated by convolving the upsampled signals with a real-valued trainable filter according to $\boldsymbol{s} = \boldsymbol{u}^\top * \boldsymbol{\theta}_2^\top$, where $*$ denotes the convolution operator. To allow for the gradient computation of the trainable PS filter, we consider a Gaussian policy. To that end, a small perturbation

TABLE II: NN parameters

		NN1 $f_{\boldsymbol{\theta}_1}$			NN2 $f_{\boldsymbol{\theta}_2}$			NN3 $f_{\boldsymbol{\theta}_3}$			NN4 $f_{\boldsymbol{\theta}_4}$		
	layer	input	hidden	output	input	hidden	output	input	hidden	output	input	hidden	output
(i)	# of layers	-	2	-	-	0	-	-	2	-	-	2	-
	# of neurons per layer	M	50	2	201	-	1	1	100	1	2	50	M
	act. function	-	ReLU	Linear	-	-	Linear	-	ReLu	-	-	ReLu	Softmax

$w_k \sim \mathcal{CN}(0, \sigma^2)$ is applied to each of the pulse-shaped signals before applying the DPD. Therefore, the DPD input $\tilde{\boldsymbol{s}} = \boldsymbol{s}+\boldsymbol{w}$ is stochastic and can be described by the PDF

$$\pi_{\boldsymbol{\theta}_2}(\tilde{s}_k|\boldsymbol{u}_k^{(L2)}) = \frac{1}{2\pi\sigma^2} e^{-\frac{|\tilde{s}_k - \boldsymbol{\theta}_2^\top \boldsymbol{u}_k^{(L_1)}|^2}{2\sigma^2}}. \tag{15}$$

At the receiver, the channel observations $\hat{\boldsymbol{y}} = [\hat{y}_1, \ldots, \hat{y}_{|\mathcal{B}_t|R}]$ are filtered by a MF and then downsampled with rate R. Then, the resulting signals $\hat{\boldsymbol{x}} = [\hat{x}_1, \ldots, \hat{x}_{|\mathcal{B}_t|}]$ are used to compute the per-example loss defined by

$$\ell_k = \log[f_{\boldsymbol{\theta}_4}(\hat{x}_k)]_{m_k}, \quad k = 1, \ldots, |\mathcal{B}_t|, \tag{16}$$

where $\hat{x}_k = \hat{u}_{kR}$. The per-example losses are sent back to the transmitter to perform the PS filter training. Due to memory effects introduced by the convolution operation in the MF and PS filter, ℓ_k is related to a subset of the entire sequence $\boldsymbol{x}$ and $\boldsymbol{m}$. We denote the total number of samples related to ℓ_k by $2G+1$. The training objective is to optimize $\boldsymbol{\theta}_2$ such that the expected cross-entropy loss $\mathcal{L}(\boldsymbol{\theta}_2) = \mathbb{E}\{\ell_k\}$ is minimized. Following [23], [34], we compute $\nabla_{\boldsymbol{\theta}_2}\mathcal{L}(\boldsymbol{\theta}_2)$ using the following proposition.

Proposition 1: The gradient of $\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}_2)$ can be approximated by

$$\begin{aligned}&\nabla_{\boldsymbol{\theta}_2}\mathcal{L}_{\mathcal{B}_t}(\boldsymbol{\theta}_2) \qquad (17)\\ &\approx \frac{1}{\sigma^2}\sum_{g=-G}^{G}\sum_{k=1}^{|\mathcal{B}_t|}\frac{1}{|\mathcal{B}_t|}\ell_k(\hat{\boldsymbol{y}}, m_k)(\tilde{s}_{kR+g} - [\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m}))]_{kR+g})\\ &\times \nabla_{\boldsymbol{\theta}_2}[\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m}))]_{kR+g},\end{aligned}$$

where we wrote $\boldsymbol{f}_{\boldsymbol{\theta}_i}$ to highlight that the relation is applied to the entire sequence in order to generate the entire corresponding output.

Proof: See Appendix A.

V. Numerical Results

In this section, we provide extensive numerical results to verify and illustrate the effectiveness of the proposed AE-based WDM system. The system performance is measured in terms of SER, and for all the results presented below, the MF used is the root-raised-cosine (RRC) filter.

A. Setup and Parameters

1) Simulation setup: We set $M = 64$, and consider a single channel system as well as a WDM system with 3 channels. For the 3-channel setup, the guard band between the adjacent channels is ηf_{b} (i.e., the channel spacing between neighboring channels is $(1+\eta)f_b$), where $\eta \geq 0$ and f_{b} is the symbol rate. The oversampling rate is set to $R = 2$ except for part of Section V-B2, where we study the impact of the oversampling rate on the performance. Both the PS filter and the MF have 201 taps. The hardware impairments considered in this paper are restricted to the IQM nonlinearity and the limited ENOB of the DAC, while the PA is assumed to be linear, as the PA nonlinearity is negligible when compared to that of the MZM. However, it should be noted that the proposed approach can be readily applied to a more general setup where the other transmitter components are not idealized (e.g., nonlinear PA and bandwidth-limited DAC (see our previous work [50]).

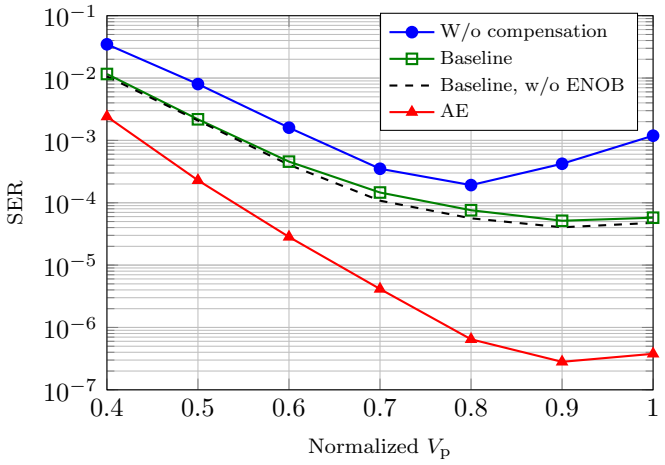

Fig. 4: (a): SER performance versus V_{p} for the single channel scenario with $\beta = 10\%$, the blue curve corresponds to the baseline setup but without applying the *arcsin* and clipping based DPD.

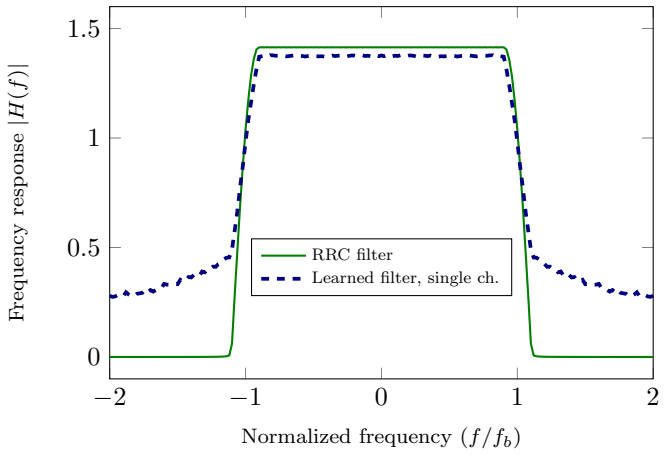

Fig. 5: Frequency response of the filter learned in the single-channel setup, showing OOB. The modulator driving swing is $V_{\mathrm{p}} = 1$ and the receiver MF roll-off factor is set to $\beta = 10\%$. The frequency response of the RRC filter with $\beta = 10\%$ roll-off is also shown as a reference.

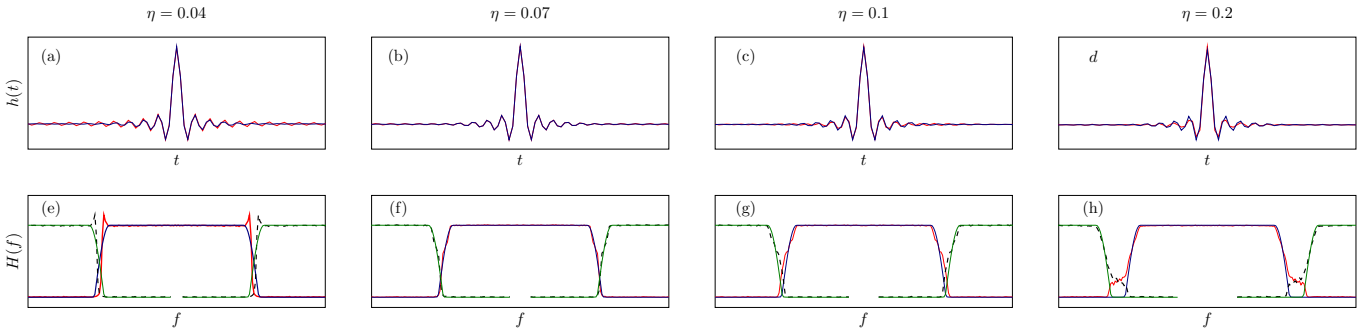

Fig. 6: The impulse (top) and frequency (bottom) response of the learned filter (red curve) versus the guard band bandwidth for $R = 2$, $V_{\mathrm{p}} = 1$ and $\beta = 10\%$. The impulse and frequency response of the RRC filter (blue) with $\beta = 10\%$ are also shown as references; The green solid and black dashed curve correspond to the RRC filter and the learned filter of the adjacent channels.

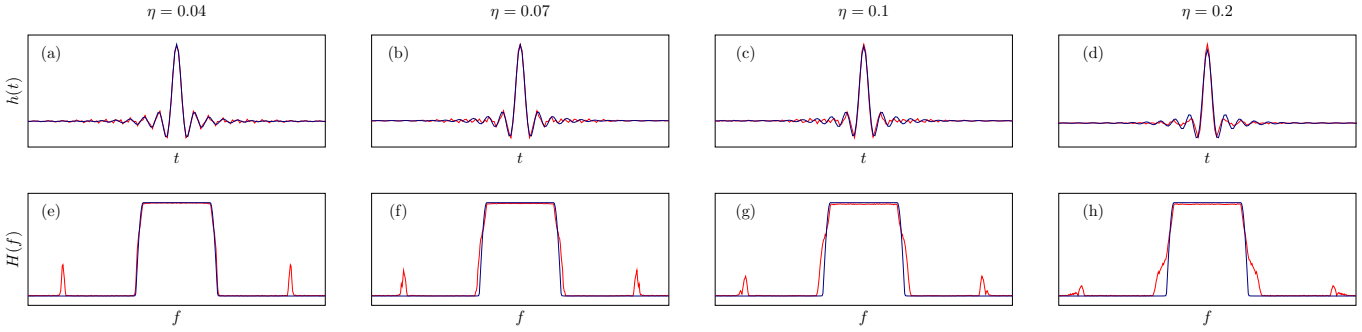

Fig. 7: The impulse (top) and frequency (bottom) response of the filters learned in a WDM system with 5 channels for $R = 4$, $V_{\mathrm{p}} = 1$ and $\beta = 10\%$. The impulse and frequency response of the RRC filter (blue) with $\beta = 10\%$ are also shown as references.

2) Transmitter and Receiver Networks: Following previous work, all NNs are implemented as multi-layer fully-connected NNs, where the ReLU function is chosen as the activation function for the hidden layers. It should be noted that other activation functions such as ELU, leaky ReLu, and Sigmoid linear unit can also be used, while they are generally more computationally expensive and therefore not considered in this paper. The NN parameters used in this paper are summarized in Table II.[7]

3) Training: All AEs are trained by minimizing the end-to-end cross-entropy loss, with the learning rate and batch size set to 0.0002 and 16000, respectively. For the 3-channel setup in particular, we consider using the same AE configuration for all 3 channels, and we therefore only minimize the cross-entropy loss of the center channel and then use the parameters of the center channel AE for the AEs of the side channels. All AEs are trained for 10000 training iterations, where one gradient update is performed in each iteration. In each training iteration, 16000 uniformly distributed training data are randomly generated, and a total number of 1.6×10^8 data samples are used for each AE optimization. For the performance evaluation, in order to avoid leakage of training data into the testing set, independent uniformly distributed data are randomly generated for testing.

[7]To optimize the number of hidden layers and the number of neurons per layer, several AEs with different sizes are trained, and we choose the AE with the best performance and relatively small number of trainable parameters. We remark that it is generally difficult to claim the chosen NN configurations are globally optimal due to large searching space.

4) Baseline: For the baseline, we use a geometrically shaped constellation that is obtained via training a standard AE [11] over an AWGN channel at SNR $= 18\,\mathrm{dB}$.[8] The PS filter is chosen as the RRC filter with roll-off factor β, which is the same as the MF at the receiver. The DPD, which operates separately on the in-phase and quadrature components, is based on the *arcsin* operation combined with clipping that can be described as [51]

$$\tilde{s}_k = \begin{cases} \min(\frac{\pi}{2}, V_{\text{clip}} \arcsin(s_k)) & s_k \geq 0 \\ \max(-\frac{\pi}{2}, V_{\text{clip}} \arcsin(s_k)) & s_k < 0, \end{cases} \tag{18}$$

where the $\arcsin$ linearizes the IQM response, while the clipping factor V_{clip} needs to be optimized to reduce the peak-to-average power ratio.

B. Results and Discussion

1) Single-Channel System: We start by investigating a single-channel scenario (e.g., there is no ICI in the system), and we evaluate the performance of the proposed method with respect to the peak voltage V_{p} of the driving signals. For notation convenience, the peak voltage of driving signals is normalized and the full swing of the MZM is used if $V_{\mathrm{p}} = 1$. Due to the dependence of the MZM nonlinearity level on the driving voltage swing, a separate AE is trained

[8]It is found that training the standard AE at SNR $= 18\,\mathrm{dB}$ leads to a constellation that is more performant than the standard square 64-QAM for a range of SNRs from $0\,\mathrm{dB}$ to $26\,\mathrm{dB}$ over the AWGN channel. The SNR of the considered WDM system falls into this SNR regime.

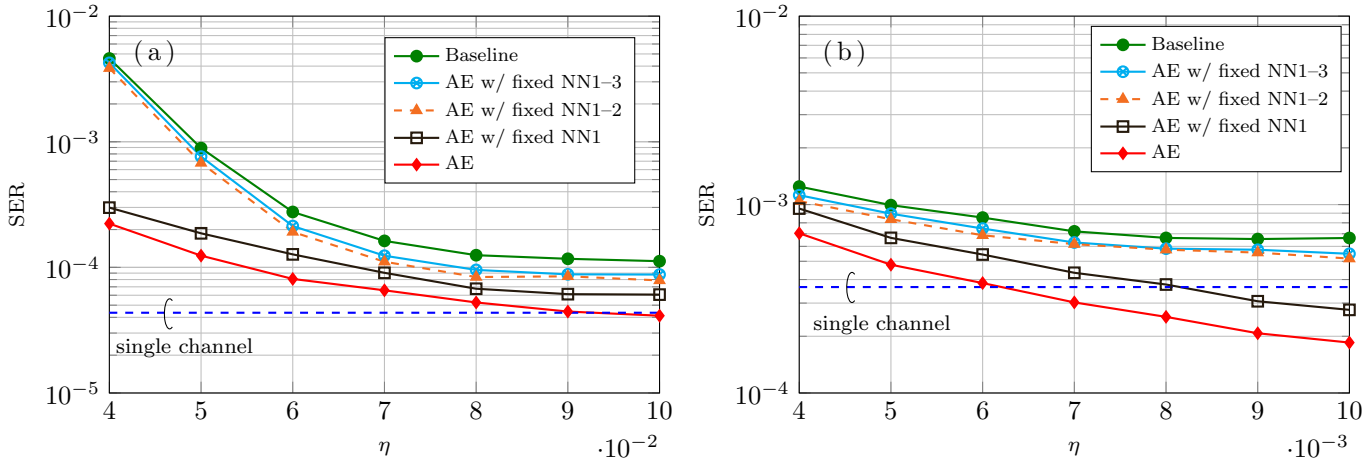

Fig. 8: SER performance versus V_{p} for the 3-channel scenario with $\beta = 10\%$ (left) and $\beta = 1\%$ (right) , the dashed blue curve corresponds to the baseline scheme for the single-channel scenario.

for each considered V_{p}. Fig. 4 visualizes the SER of the proposed system when the receiver MF roll-off factor is set to $\beta = 10\%$. For a range of considered V_{p}, the proposed approach achieves significantly better performance than the considered baseline. However, by looking at the frequency response of the learned PS filter, as shown with the blue dashed curve in Fig. 5, we observe that compared to the RRC filter with 10% roll-off, the learned filter has a significant amount of OOB energy, which will introduce severe ICI between narrowly-spaced neighboring channels and make it unsuitable for high SE WDM systems. This result indicates that the system designed for the single-channel setup cannot always be directly applied to a multi-channel setup, and additional care should be taken when designing multi-channel systems.

2) WDM System With 3 channels: We now train the proposed AE in a 3-channel setup. Fig. 6 visualizes the filters learned with different guard band bandwidth. We start by looking at the impulse response of the learned filters, which appears to be very similar to the RRC filter. However, from the frequency responses we observe that the trainable filter learns to adjust its bandwidth according to the guard band between the neighboring channels. In particular, when the guard band is small (e.g., $\eta = 0.04$, Fig. 6 (e)) the filter learns to restrict the OOB energy and has a narrower frequency response than the RRC filter, indicating that the trainable filter learns to limit ICI. As we increase the guard band bandwidth, the bandwidth of the trainable filter increases as well. Similar to the single-channel scenario, the filter learns to put a significant amount of energy in the unoccupied spectrum when the guard band is large (see Fig. 6 (h) for $\eta = 0.2$).

To train the multi-channel system it is necessary to use high oversampling rates to allow for placing the neighboring channels in the considered spectrum. We emphasize that, in this scenario, it is important to ensure that the PS filter cannot generate unrealistically high frequency components. This is illustrated in Fig. 7, which depicts the learned filters when the filter is trained with 5 channels and $R = 4$ times oversampling rate. Similar as before, the trainable filter learns to adjust its bandwidth according to the channel spacing. However, the filter also learns to put energy at high frequencies at the edges between the next two channels. Despite this interesting behavior, such a filter is not feasible in practice due to the fact that a practical system would not operate at such high sampling rate because of the hardware limitations as well as power constraints. This result reminds us again the importance of using realistic setups when applying DL techniques for designing communication systems. Instead of upsampling to the final oversampling rate before the PS, one should use $R = 2$ times oversampling rate for the PS and another upsampling step after the PS, which is the approach we followed for the other multi-channel simulations. An additional benefit of this method is that the number of filter taps is reduced for the same FIR filter length, which improves convergence.

We now evaluate the performance of the proposed system versus different guard band bandwidth, and we consider setting $R = 2$ and the receiver MF roll-off factor to $\beta = 10\%$ and $\beta = 1\%$. The achieved SER for the center channel is shown in Fig. 8 (a) for $\beta = 10\%$ and in Fig. 8 (b) for $\beta = 1\%$.[9] As a reference, the SER performance of the baseline scheme applying *arcsin* combined with clipping is also shown. We remark that the clipping factor V_{clip} and V_{p} are optimized for the baseline scheme, while V_{p} is set to 1 in the proposed scheme for simplicity. Potentially, the performance of the proposed scheme can be further improved by optimizing V_{p}—the optimal performance for the single-channel case is achieved at $V_{\mathrm{p}} = 0.9$ (see Fig. 4). For roll-off factors of 10% (Fig. 8 (a)) and 1% (Fig. 8 (b)), the proposed approach outperforms the baseline scheme over all considered guard bands. More importantly, compared to the baseline scheme, the guard band for the proposed scheme can be significantly reduced with limited impact on the SER performance —for the target SER where the baseline performance starts to saturate, the guard band can be reduced by around 37% for 10% roll-off and around 50% for 1% roll-off. Such results indicate that the proposed approach can improve the SE of WDM systems by allowing to put the channels at a very narrow channel spacing.

[9]We note that the side channels have better SER performance than the center channel as they suffer from less ICI.

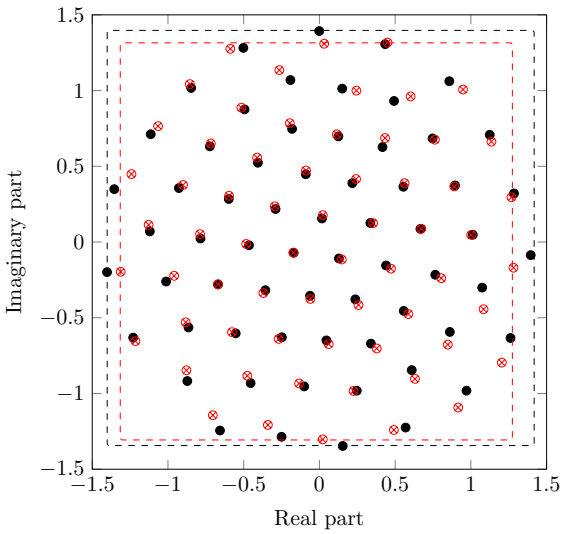

Fig. 9: Constellation used for the baseline (black) and constellation learned in the 3-channel setup with $\eta = 0.05$ (red).

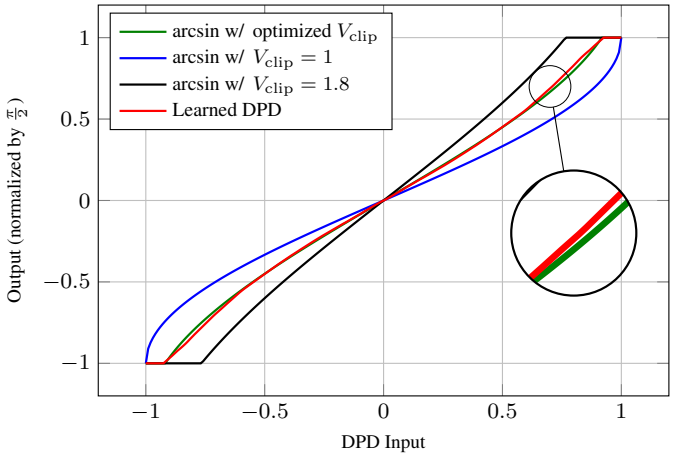

Fig. 10: Transfer function of NN3 and the conventional DPD using *arcsin* combined with optimized clipping for $\beta = 10\%$ and $\eta = 0.05$. The transfer functions of the conventional DPD using *arcsin* and sub-optimal clippings are also shown as references.

However, it should be noted that the reduction in guard bands does not translate directly into the same gain in terms of SE, as the explicit SE depends on the applied modulation formats, the channel spacing, and the resulting SER.

3) Learned Constellation: Fig. 9 visualizes the learned constellation when the AE is trained in the 3-channel setup with $\beta = 10\%$ and $\eta = 0.05$. The constellation optimized over the AWGN channel and used for the baseline is also shown as a reference. It is shown that the constellation optimized over the WDM setup has lower peak amplitude than the baseline, indicating the constellation optimized for the AWGN channel is suboptimal for a system that is impaired by hardware imperfections. One possible explanation for such observation is that the AE learns to limit the peak voltage V_p by restricting the maximum amplitude of the constellation, so as to limit the signal distortion caused by the nonlinear MZM. And the learned constellation in return highlights the importance of considering the maximum constellation amplitude when designing constellation for nonlinear systems.

4) Learned DPD: Fig. 10 visualizes the transfer function of the DPD (i.e., ANN3) learned for the 3-channel system with $\beta = 10\%$ and $\eta = 0.05$. The transfer functions of the conventional DPD employing *arcsin* and different clipping V_{clip} are also shown as references. It is shown that the baseline DPD with optimized clipping has a response similar to the learned DPD, suggesting that the considered DPD applying *arcsin* combined with optimized clipping is near optimal for the considered scenario.

5) Ablation study: In order to quantify the origin of the performance gains, we carry out an ablation study by first freezing all the pre-trained NNs and then individually unfreezing them in the order of NN4, NN3, NN2, and NN1. We start by unfreezing NN4. The resulting SER performance for $\beta = 10\%$ and $\beta = 1\%$ is shown in Fig. 8 (a) and Fig. 8 (b), respectively. Compared to the baseline scheme, it can be seen that the proposed approach achieves slightly better performance. Such result is what one would have expected, as the demampper trained over the AWGN channel is likely to be suboptimal for a channel impaired by hardware imperfections. We then further unfreeze the parameters of NN3 (i.e., NN1–2 are frozen). The resulting performance is very similar (slightly better) to the case where NN1–3 are frozen. This result is consistent with what is shown in Fig. 10, that the model-based DPD with optimized V_{clip} achieves similar performance as the NN-based DPD. Finally, the parameters of NN2 are also made trainable (i.e., only NN1 is fixed). In this case, the SER of the proposed approach improves significantly. Particularly, the largest gain achieved for $\beta = 10\%$ is $\eta = 0.04$ while is $\eta = 0.01$ for $\beta = 1\%$, indicating that the guard band can be optimized to improve the system performance. Finally, when all NNs are made trainable, the performance of the proposed method further improves, which is consistent with what is shown in Section. V-B3. Additionally, we note that the major contributor of performance improvement differs in different scenarios as it is shown in Fig. 8. This indicates the different functional blocks may have different priorities when one try to improve the communication design, depending on the considered scenarios.

We remark that the ablation study performed here is only possible when the AE design follows the architecture of conventional model-based communication systems. Such model-based AE design has the benefit of increased interpretability compared to the conventional AE-based systems, as it allows us to measure the performance of each transmitter/receiver blocks separately. However, it should be noted that it is in general hard to claim that the learned functions can be easily separated from each other.

C. Model-Free Training of the Pulse-Shaping Filter

In this section, we extend our results to the case where a differentiable channel model is unknown. Here, we only consider training of the PS filter with the generalized training algorithm discussed in Section IV-C. The reason for only learning the PS filter is that PS filter training contributes to most of the performance gain as it is shown in the ablation

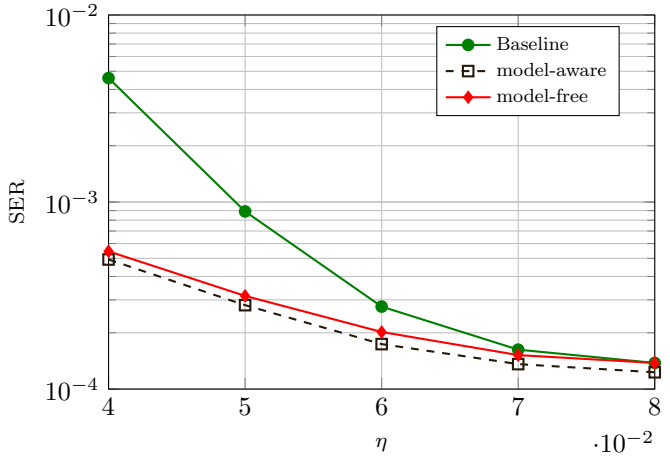

Fig. 11: SER performance comparison for $\beta = 10\%$ when the proposed AE is trained with and without the perfect channel knowledge.

study. RL-based training of the mapper and the DPD can be found in [23], [34], and [50], respectively.

Fig. 11 shows the achieved SER of the different schemes over a 3-channel WDM system. It is observed that the learned PS filter using the RL-based algorithm achieves very similar performance to the one using standard end-to-end learning assuming perfect channel knowledge. However, it should be noted that the RL-based approach allows for training of NNs in an experimental channel, and it has the potential to exceed the performance of the conventional end-to-end learning-based approach, as the performance of the latter is highly dependent on the accuracy of the model used for training.

VI. Conclusion and Future Work

We proposed a novel end-to-end AE for WDM systems that are impaired with non-ideal hardware components. In contrast to most of the conventional AEs, which are usually implemented as a pair of NNs, our AE design follows the architecture of conventional communication systems, and our transmitter is implemented by a concatenation of simple NNs. Simulation results show that the proposed AE-based system achieves significantly better performance than the considered baseline, and allows to increase the SE of WDM systems by reducing the channel spacing without severe SER performance degradation. By means of an ablation study, we quantify the origin of the performance improvement. It is shown that the performance gain can be ascribed to the optimized constellation mapper, PS filter, and demapper. In addition, in case the channel model is unknown, we have shown that the PS filter can be trained using RL, and our simulation results indicate that the extended RL-based training approach can achieve similar performance to the standard end-to-end learning assuming perfect channel knowledge.

For future work, there are several important aspects concerning the use of AEs which deserve further study:

- Channel models: We have considered an optical back-to-back channel due to the fact that the hardware distortions alone significantly degrade the system. However, practical systems further suffer from performance loss caused by the nonlinear crosstalk between adjacent channels. The AE-based method may help to reduce the impact of the crosstalk and provide significant performance improvement.
- The current AE design assumes that the WDM channels operate at the same rate. Practical systems, however, allow for transmission at different rates. New AE design and training methods may be needed to allow for flexible transmission rates.

Appendix

We work with complete sequences, so that the loss is given by:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}_2) &= \mathbb{E}_{\boldsymbol{m},\boldsymbol{s},\tilde{\boldsymbol{s}},\boldsymbol{y},\hat{\boldsymbol{y}},\hat{\boldsymbol{x}}}\{\ell_k\} \\ &= \mathbb{E}_{\boldsymbol{m},\tilde{\boldsymbol{s}}|\boldsymbol{m},\hat{\boldsymbol{y}}|\tilde{\boldsymbol{s}}}\{\ell_k\}\end{aligned} \tag{19}$$

where in the second step we remove all the deterministic relations. Hence

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}_2) &= \sum_{\boldsymbol{m}} \iint p(\boldsymbol{m})p(\tilde{\boldsymbol{s}}|\boldsymbol{m})p(\hat{\boldsymbol{y}}|\tilde{\boldsymbol{s}})\ell_k(\hat{\boldsymbol{y}},m_k)\mathrm{d}\tilde{\boldsymbol{s}}\mathrm{d}\hat{\boldsymbol{y}} \\ &= \sum_{\boldsymbol{m}} \iint p(\boldsymbol{m})\pi(\tilde{\boldsymbol{s}}|\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m})))p(\hat{\boldsymbol{y}}|\tilde{\boldsymbol{s}})\ell_k(\hat{\boldsymbol{y}},m_k)\mathrm{d}\tilde{\boldsymbol{s}}\mathrm{d}\hat{\boldsymbol{y}},\end{aligned} \tag{20}$$

where we wrote $\boldsymbol{f}_{\boldsymbol{\theta}_i}$ to expressly denote that the relation is applied to the entire sequence in order to generate entire the corresponding output. Exploiting the policy gradient theorem [34] and using the fact that $\nabla_x \log(g(x)) = \frac{\nabla_x g(x)}{g(x)}$, it then follows that

$$\begin{aligned}&\nabla_{\boldsymbol{\theta}_2}\mathcal{L}(\boldsymbol{\theta}_2) \\ &= \sum_{\boldsymbol{m}} \iint p(\boldsymbol{m})\nabla_{\boldsymbol{\theta}_2}\pi(\tilde{\boldsymbol{s}}|\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m})))p(\hat{\boldsymbol{y}}|\tilde{\boldsymbol{s}})\ell_k(\hat{\boldsymbol{y}},m_k)\mathrm{d}\tilde{\boldsymbol{s}}\mathrm{d}\hat{\boldsymbol{y}} \\ &= \mathbb{E}\{\ell_k(\hat{\boldsymbol{y}},m_k)\nabla_{\boldsymbol{\theta}_2}\log\pi(\tilde{\boldsymbol{s}}|\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m})))\} \\ &= \mathbb{E}\{\ell_k(\hat{\boldsymbol{y}},m_k)\sum_{i=1}^{R|\mathcal{B}_t|}\nabla_{\boldsymbol{\theta}_2}\log\pi(\tilde{s}_i|[\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m}))]_i)\} \\ &\approx \mathbb{E}\{\ell_k(\hat{\boldsymbol{y}},m_k)\sum_{g=-G}^{G}\nabla_{\boldsymbol{\theta}_2}\log\pi(\tilde{s}_{kR+g}|[\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m}))]_{kR+g})\} \\ &= \frac{1}{\sigma^2}\sum_{g=-G}^{G}\mathbb{E}\{\ell_k(\hat{\boldsymbol{y}},m_k)(\tilde{s}_{kR+g}-[\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m}))]_{kR+g}) \\ &\times\nabla_{\boldsymbol{\theta}_2}[\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m}))]_{kR+g}\} \\ &\approx \frac{1}{\sigma^2}\sum_{g=-G}^{G}\sum_{k=1}^{|\mathcal{B}_t|}\frac{1}{|\mathcal{B}_t|}\ell_k(\hat{\boldsymbol{y}},m_k)(\tilde{s}_{kR+g}-[\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m}))]_{kR+g}) \\ &\times\nabla_{\boldsymbol{\theta}_2}[\boldsymbol{f}_{\boldsymbol{\theta}_2}(\boldsymbol{f}_{\boldsymbol{\theta}_1}(\boldsymbol{m}))]_{kR+g},\end{aligned}$$

which leads us to (17). The first approximation considers that $\ell_k(\hat{\boldsymbol{y}},m_k)$ is only affected by $2G+1$ surrounding samples, while the second approximation is used to compute the expectation by averaging over the batch. We ignored boundary effect at the start and end of the sequence.

REFERENCES

[1] J. Song, C. Häger, J. Schröder, A. Graell i Amat, and H. Wymeersch, "End-to-end autoencoder for superchannel transceivers with hardware impairment," in *Proc. Optical Fiber Communications Conference (OFC)*. IEEE, 2021, p. F4D.6.

[2] P. J. Winzer, D. T. Neilson, and A. R. Chraplyvy, "Fiber-optic transmission and networking: the previous 20 and the next 20 years," *Optics express*, vol. 26, no. 18, pp. 24 190–24 239, 2018.

[3] M. Yamada, A. Mori, K. Kobayashi, H. Ono, T. Kanamori, K. Oikawa, Y. Nishida, and Y. Ohishi, "Gain-flattened tellurite-based EDFA with a flat amplification bandwidth of 76 nm," *IEEE Photonics Technology Letters*, vol. 10, no. 9, pp. 1244–1246, 1998.

[4] D. Rafique, T. Rahman, A. Napoli, M. Kuschnerov, G. Lehmann, and B. Spinnler, "Flex-grid optical networks: spectrum allocation and nonlinear dynamics of super-channels," *Optics Express*, vol. 21, no. 26, pp. 32 184–32 191, 2013.

[5] M. Mazur, J. Schröder, M. Karlsson, and P. A. Andrekson, "Joint superchannel digital signal processing for ultimate bandwidth utilization," *arXiv preprint arXiv:1911.02326*, 2019.

[6] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, "Machine learning paradigms for next-generation wireless networks," *Wireless Communications*, vol. 24, no. 2, pp. 98–105, 2016.

[7] F. N. Khan, C. Lu, and A. P. T. Lau, "Machine learning methods for optical communication systems," in *Proc. Signal Processing in Photonic Communications*, 2017, pp. SpW2F–3.

[8] N. E. West and T. O'Shea, "Deep architectures for modulation recognition," in *Proc. International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2017, pp. 1–6.

[9] D. Zibar, M. Piels, R. Jones, and C. G. Schäeffer, "Machine learning techniques in optical communication," *Journal of Lightwave Technology*, vol. 34, no. 6, pp. 1442–1452, 2015.

[10] C. Häger and H. D. Pfister, "Nonlinear interference mitigation via deep neural networks," in *Proc. Optical Fiber Communications Conference (OFC)*, no. W3A.4., 2018.

[11] T. O'shea and J. Hoydis, "An introduction to deep learning for the physical layer," *Transactions on Cognitive Communications and Networking*, vol. 3, no. 4, pp. 563–575, 2017.

[12] S. Dörner, S. Cammerer, J. Hoydis, and S. Ten Brink, "Deep learning based communication over the air," *Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 132–143, 2017.

[13] A. Felix, S. Cammerer, S. Dörner, J. Hoydis, and S. Ten Brink, "OFDM-autoencoder for end-to-end learning of communications systems," in *Proc. International Workshop on Signal Processing Advances in Wireless Communications*, 2018, pp. 1–5.

[14] S. Li, C. Häger, N. Garcia, and H. Wymeersch, "Achievable information rates for nonlinear fiber communication via end-to-end autoencoder learning," in *Proc. European Conference on Optical Communication (ECOC)*, 2018, pp. 1–3.

[15] R. T. Jones *et al.*, "Geometric constellation shaping for fiber optic communication systems via end-to-end learning," *arXiv preprint arXiv:1810.00774*, 2018.

[16] 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 *Proc. European Conference on Optical Communication (ECOC)*. IEEE, 2021, pp. 1–4.

[17] R. T. Jones, T. A. Eriksson, M. P. Yankov, B. J. Puttnam, G. Rademacher, R. S. Luis, and D. Zibar, "End-to-end deep learning of optical fiber communications," *Journal of Lightwave Technology*, vol. 36, no. 20, pp. 4843–4855, 2018.

[18] T. Uhlemann, S. Cammerer, A. Span, S. Dörner, and S. ten Brink, "Deep-learning autoencoder for coherent and nonlinear optical communication," in *Proc. ITG-Symposium on Photonic Networks*, 2020, pp. 1–8.

[19] F. A. Aoudia and J. Hoydis, "Waveform learning for next-generation wireless communication systems," *arXiv preprint arXiv:2109.00998*, 2021.

[20] M. Kim, W. Lee, and D.-H. Cho, "A novel PAPR reduction scheme for OFDM system based on deep learning," *Communications Letters*, vol. 22, no. 3, pp. 510–513, 2017.

[21] T. J. O'Shea, T. Erpek, and T. C. Clancy, "Physical layer deep learning of encodings for the MIMO fading channel," in *Proc. Annual Allerton Conference on Communication, Control, and Computing*, 2017, pp. 76–80.

[22] H. Ye, G. Y. Li, B.-H. F. Juang, and K. Sivanesan, "Channel agnostic end-to-end learning based communication systems with conditional GAN," in *Proc. Globecom Workshops*, 2018, pp. 1–5.

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

[24] M. Zhang, M. Liu, and Z. Zhong, "Neural network assisted active constellation extension for papr reduction of OFDM system," in *Proc. International Conference on Wireless Communications and Signal Processing*, 2019, pp. 1–5.

[25] K. Choi, K. Tatwawadi, A. Grover, T. Weissman, and S. Ermon, "Neural joint source-channel coding," in *Proc. International Conference on Machine Learning*, 2019, pp. 1182–1192.

[26] M. Stark, F. A. Aoudia, and J. Hoydis, "Joint learning of geometric and probabilistic constellation shaping," in *Proc. Globecom Workshops*, 2019.

[27] S. Cammerer, F. A. Aoudia, S. Dörner, M. Stark, J. Hoydis, and S. Ten Brink, "Trainable communication systems: Concepts and prototype," *Transactions on Communications*, vol. 68, no. 9, pp. 5489–5503, 2020.

[28] T. Van Luong, Y. Ko, M. Matthaiou, N. A. Vien, M.-T. Le, and V.-D. Ngo, "Deep learning-aided multicarrier systems," *Transactions on Wireless Communications*, vol. 20, no. 3, pp. 2109–2119, 2020.

[29] F. A. Aoudia and J. Hoydis, "End-to-end learning for OFDM: From neural receivers to pilotless communication," *Transactions on Wireless Communications*, 2021.

[30] R. T. Jones, M. P. Yankov, and D. Zibar, "End-to-end learning for GMI optimized geometric constellation shape," in *Proc. European Conference on Optical Communication (ECOC)*, 2019, pp. 1–3.

[31] B. Karanov, D. Lavery, P. Bayvel, and L. Schmalen, "End-to-end optimized transmission over dispersive intensity-modulated channels using bidirectional recurrent neural networks," *Optics express*, vol. 27, no. 14, pp. 19 650–19 663, 2019.

[32] K. Gümüş, A. Alvarado, B. Chen, C. Häger, and E. Agrell, "End-to-end learning of geometrical shaping maximizing generalized mutual information," in *Proc. Optical Fiber Communications Conference (OFC)*, 2020, p. W3D.4.

[33] B. Karanov, V. Oliari, M. Chagnon, G. Liga, A. Alvarado, V. Aref, D. Lavery, P. Bayvel, and L. Schmalen, "End-to-end learning in optical fiber communications: Experimental demonstration and future trends," in *Proc. European Conference on Optical Communications (ECOC)*, 2020, pp. 1–3.

[34] F. A. Aoudia and J. Hoydis, "Model-free training of end-to-end communication systems," *Journal on Selected Areas in Communications*, vol. 37, no. 11, pp. 2503–2516, 2019.

[35] A. C. Wilson, R. Roelofs, M. Stern, N. Srebro, and B. Recht, "The marginal value of adaptive gradient methods in machine learning," *arXiv preprint arXiv:1705.08292*, 2017.

[36] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[37] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[38] P. a. Rodríguez, "Beyond one-hot encoding: Lower dimensional target embedding," *Image and Vision Computing*, vol. 75, pp. 21–31, 2018.

[39] M. Li, D. Wang, Q. Cui, Z. Zhang, L. Deng, and M. Zhang, "End-to-end learning for optical fiber communication with data-driven channel model," in *Proc. Opto-Electronics and Communications Conference*, 2020, pp. 1–3.

[40] D. Wang, Y. Song, J. Li, J. Qin, T. Yang, M. Zhang, X. Chen, and A. C. Boucouvalas, "Data-driven optical fiber channel modeling: a deep learning approach," *Journal of Lightwave Technology*, vol. 38, no. 17, pp. 4730–4743, 2020.

[41] B. Karanov, M. Chagnon, V. Aref, D. Lavery, P. Bayvel, and L. Schmalen, "Concept and experimental demonstration of optical im/dd end-to-end system optimization using a generative model," in *Proc. Optical Fiber Communications Conference (OFC)*, 2020, pp. 1–3.

[42] V. Curri, A. Carena, G. Bosco, P. Poggiolini, and F. Forghieri, "Optimization of DSP-based Nyquist-WDM PM-16QAM transmitter," in *Proc. European Conference on Optical Communication (ECOC)*. Optical Society of America, 2012, pp. 1–3.

[43] G. Khanna, B. Spinnler, S. Calabrò, E. De Man, and N. Hanik, "A robust adaptive pre-distortion method for optical communication transmitters," *Photonics Technology Letters*, vol. 28, no. 7, pp. 752–755, 2015.

[44] P. W. Berenguer, M. Nölle, L. Molle, T. Raman, A. Napoli, C. Schubert, and J. K. Fischer, "Nonlinear digital pre-distortion of transmitter components," *Journal of lightwave technology*, vol. 34, no. 8, pp. 1739–1745, 2015.

[45] A. Napoli, P. W. Berenguer, T. Rahman, G. Khanna, M. M. Mezghanni, L. Gardian, E. Riccardi, A. C. Piat, S. Calabrò, S. Dris *et al.*, "Digital pre-compensation techniques enabling high-capacity bandwidth variable transponders," *Optics Communications*, vol. 409, pp. 52–65, 2018.

[46] C. Laperle and M. O'Sullivan, "Advances in high-speed DACs, ADCs, and DSP for optical coherent transceivers," *Journal of lightwave technology*, vol. 32, no. 4, pp. 629–643, 2014.
[47] A. Napoli, M. M. Mezghanni, T. Rahman, D. Rafique, R. Palmer, B. Spinnler, S. Calabrò, C. Castro, M. Kuschnerov, and M. Bohn, "Digital compensation of bandwidth limitations for high-speed DACs and ADCs," *Journal of Lightwave Technology*, vol. 34, no. 13, pp. 3053–3064, 2016.
[48] D. Mishkin and J. Matas, "All you need is a good init," *arXiv preprint arXiv:1511.06422*, 2015.
[49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[50] J. Song, Z. He, C. Häger, M. Karlsson, A. Graell i Amat, H. Wymeersch, and J. Schröder, "Over-the-fiber digital predistortion using reinforcement learning," in *Proc European Conference on Optical Communication (ECOC)*. IEEE, 2021, pp. 1–4.
[51] Y. Tang, K.-P. Ho, and W. Shieh, "Coherent optical OFDM transmitter design employing predistortion," *Photonics Technology Letter*, vol. 20, no. 11, pp. 954–956, 2008.

Jinxiang Song (S'20) obtained the M.S.c degree in Electrical Engineering from Chalmers University of Technology, Gothenburg, Sweden, in 2019, where he is currently pursuing the Ph.D. degree in the Department of Electrical Engineering. His main research interests lie in digital communication, machine learning and signal processing.

Christian Häger received the Dipl.-Ing. degree (M.Sc. equivalent) from Ulm University, Germany, in 2011 and his Ph.D. degree from Chalmers University of Technology, Sweden, in 2016. He is currently an Assistant Professor in the Department of Electrical Engineering at Chalmers University of Technology, Sweden. Before that, he was a postdoctoral researcher at the Department of Electrical and Computer Engineering at Duke University, USA and at the Department of Electrical Engineering at Chalmers University of Technology. His research interests lie at the intersection of communication systems, machine learning, and signal processing. He received the Marie Sklodowska-Curie Global Fellowship from the European Commission in 2017 and a Starting Grant from the Swedish Research Council in 2020.

Jochen Schröder (M'2010) is a Senior Researcher (tenured) in the Photonics Laboratory at the Department of Microtechnology and Nanoscience at the Chalmers University of Technology in Gothenburg Sweden. He graduated with a Ph.D. from the University of Auckland in 2010 and subsequently worked as a Postdoc and Senior Research Fellow at the Centre for Ultrahighbandwidth Devices for Optical Systems (CUDOS) at the University of Sydney. During his time in Sydney he held an Australian Research Council Discovery Early Career Research Award and was the recipient of the Australian Optical Society Geoff Opat Early Career Researcher Prize and a finalist of the Australian Museum Eureka Prize. He has authored more than 150 journal and conference publications including postdeadline presentations at ECOC and OFC.

Alexandre Graell i Amat (S'01–M'05–SM'10) is a Professor at the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden. He received the M.Sc. degree in Telecommunications Engineering from the Universitat Politécnica de Catalunya, Barcelona, Catalonia, Spain, in 2001, and the M.Sc. and Ph.D. degrees in Electrical Engineering from the Politecnico di Torino, Turin, Italy, in 2000 and 2004, respectively. From 2001 to 2002, he was a Visiting Scholar with the University of California San Diego, La Jolla, CA, USA. From 2002 to 2003, he held a visiting appointment at Universitat Pompeu Fabra, Barcelona, and the Telecommunications Technological Center of Catalonia, Barcelona. From 2001 to 2004, he held a part-time appointment at the STMicroelectronics Data Storage Division, Milan, Italy, as a consultant on coding for magnetic recording channels. From 2004 to 2005, he was a Visiting Professor with Universitat Pompeu Fabra, Barcelona. From 2006 to 2010, he was with the Department of Electronics, IMT Atlantique (formerly ENST Bretagne), Brest, France. Since 2019 he is also Adjunct Research Scientist at Simula UiB, Bergen, Norway. His research interests are in the field of coding theory with application to distributed computing, privacy and security, random access, and optical communications. Prof. Graell i Amat received the Marie Skłodowska-Curie Fellowship from the European Commission and the Juan de la Cierva Fellowship from the Spanish Ministry of Education and Science. He received the IEEE Communications Society 2010 Europe, Middle East, and Africa Region Outstanding Young Researcher Award. He was the General Co-Chair of the 7th International Symposium on Turbo Codes and Iterative Information Processing, Sweden, 2012, and the TPC Co-Chair of the 11th International Symposium on Topics in Coding, Canada, 2021. He was an Associate Editor of the IEEE COMMUNICATIONS LETTERS from 2011 to 2013. He was Associate Editor and Editor-at-Large of the IEEE TRANSACTIONS ON COMMUNICATIONS from 2011 to 2016 and 2017 to 2020, respectively. He is currently Area Editor of the IEEE TRANSACTIONS ON COMMUNICATIONS.

Henk Wymeersch (S'01, M'05, SM'19) obtained the Ph.D. degree in Electrical Engineering/Applied Sciences in 2005 from Ghent University, Belgium. He is currently a Professor of Communication Systems with the Department of Electrical Engineering at Chalmers University of Technology, Sweden. He is also a Distinguished Research Associate with Eindhoven University of Technology. Prior to joining Chalmers, he was a postdoctoral researcher from 2005 until 2009 with the Laboratory for Information and Decision Systems at the Massachusetts Institute of Technology. Prof. Wymeersch served as Associate Editor for IEEE Communication Letters (2009-2013), IEEE Transactions on Wireless Communications (since 2013), and IEEE Transactions on Communications (2016-2018) and is currently Senior Member of the IEEE Signal Processing Magazine Editorial Board. During 2019-2021, he was an IEEE Distinguished Lecturer with the Vehicular Technology Society. His current research interests include the convergence of communication and sensing, in a 5G and Beyond 5G context.