This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2025.3570452

Blind Equalization using a Variational Autoencoder with Second Order Volterra Channel Model

Søren Føns Nielsen, Darko Zibar and Mikkel N. Schmidt

***Abstract*—Existing communication hardware is being exerted to its limits to accommodate for the ever increasing internet usage globally. This leads to non-linear distortion in the communication link that requires non-linear equalization techniques to operate the link at a reasonable bit error rate. This paper addresses the challenge of blind non-linear equalization using a variational autoencoder (VAE) with a second-order Volterra channel model. The VAE framework's costfunction, the evidence lower bound (ELBO), is derived for real-valued constellations and can be evaluated analytically without resorting to sampling techniques. We demonstrate the effectiveness of our approach through simulations on a synthetic Wiener-Hammerstein channel and a simulated intensity modulated direct detection (IM/DD) optical link. The results show significant improvements in equalization performance, compared to a VAE with linear channel assumptions, highlighting the importance of appropriate channel modeling in unsupervised VAE equalizer frameworks.**

I. Introduction

In recent years, internet usage has increased dramatically in part due to the availability of video streaming and social media. Furthermore, the recent surge in training large machine learning models has led to the construction of large scale datacenters to support fast turnaround [1]. This means that existing communications hardware is being pushed to its limits, which in many applications leads to non-linear distortion. This could for instance be saturation effects from radio frequency power amplifiers [2], the transfer function in the light emitting diode in visible light communication [3] or non-ideal modulators, chromatic dispersion and detection in short-reach optical networks [4]. Future communication solutions need to be able to handle non-linear distortion to a larger degree than before.

The process of removing distortion and noise caused by the communication channel at the receiver is commonly known as *equalization*. One way to optimize the equalizer is using a sequence of apriori known data symbols, a *pilot* sequence (a supervised approach). For linear channel distortion and intersymbol interference (ISI) a linear adaptive filter can be used, which commonly is done either through a feed-forward filter (FFE) [5] or with a combined feed-forward and feed-back filter system, denoted a decision-feedback equalizer (DFE) [6]. However, many communication channels are subject to non-linear distortion which require a non-linear equalizer to fully compensate. Popularly, this has been tackled with a Volterra equalizer [7], due to its solid theoretical foundation and stability guarantees. The Volterra series is used to describe general non-linear systems [8] and uses a polynomial basis of past inputs to predict the next output. Its theory was developed by Norbert Wiener (however named after Vito Volterra) and has since then been used to model a variety of non-linear systems, such as the transfer function of the power amplifier in wireless communication [9], the brain's hemodynamic response function in conjuction with functional magnetic resonance imaging [10] and to approximate the rate equations of a light emitting diode (LED) used in visible light communication (VLC) [11], to mention a few.

Another class of non-linear equalizers are the *neural networks* which have also attracted some attention both during their early adoption in the 1990s [12][13] and also more recently [14][15]. Compared to the Volterra equalizers they are more flexible as they learn a basis via. composable non-linear functions (layers) directly from data, however, they can also be more difficult to train.

The *supervised* approach of sending pilot symbols decreases the throughput of the communication system and thus much effort has also gone into investigating *blind* (unsupervised) approaches. In this scheme, only knowledge of the constellation can be utilized for optimizing the equalizer weights. This was first studied for pulse amplitude modulation (PAM) formats in [16]. For complex-valued modulation formats, the most widely used algorithm in this category is the constant-modulus algorithm (CMA) [17]. It utilizes a criterion, based on the average modulus of the constellation, to optimize a finite impulse response (FIR) filter. To accommodate for non-constant modulus constellations, an extension to CMA was proposed called the multi modulus algorithm (MMA) [18].

More recently, a new class of blind equalization algorithms have been proposed based on a Bayesian formulation of the problem; first in [19], [20] for the quadrature phase shift keying (QPSK) modulation format with coded data and later extended to quadrature amplitude modulation (QAM) with probabilistic constellation shaping (PCS) in [21]. Both works are based on formulating the equalization problem as a variational autoencoder (VAE) [22], which tries to approximate the maximum a posterior (MAP) symbol sequence with a simpler distribution by maximizing the evidence lower bound (ELBO). It was shown in [20], that the ELBO has an analytical expression, which can be differentiated wrt. parameters of the model, if an FIR filter is used to model the channel. An approximation of the VAE equalizer was explored in [23], named the vector quantized variational autoencoder (VQ-VAE). Here the authors present an alternative costfunction

This work was supported by VILLUM FONDEN with grants MARBLE (VIL40555) and VI-POPCOM (VIL54486)

S. F. Nielsen and M. N. Schmidt are with the Department of Applied Mathematics and Computer Science, Technical University of Denmark mail: sfvn@dtu.dk.

Darko Zibar is with the Department of Electrical and Photonics Engineering, Technical University of Denmark

Manuscript received XXXX; revised XXXX.

Authorized licensed use limited to: Technical University of Denmark DTU Library. Downloaded on May 19,2025 at 08:41:32 UTC from IEEE Xplore. Restrictions apply.
© 2025 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2025.3570452

to the ELBO, that admits arbitrary non-linear channel model by using a low-complexity hard symbol demapper. However, the VQ-VAE no longer learns the channel equalization and demapping jointly, in contrast to the VAE from [19], [21], as the hard-demapper is fixed.

In this work, we extend the VAE framework from [20], [21] to incorporate a non-linear second order Volterra channel model for real-valued constellations. Our main contribution is the derivation of the Volterra VAE (V2VAE), including an analytical expression for the ELBO. This allows for *joint* estimation of the channel equalizer and soft symbol demapping, which is a faithful estimation of the MAP symbol decoder given a Volterra channel. We contrast our proposed model, to a baseline model from [23] that uses a simplified memory polynomial as the channel model in the VAE framework, denoted MP-VAE. We investigate our model's merits in two simulated data studies. We first show the advantage of using the non-linear channel assumptions in a Wiener-Hammerstein system. Finally, we analyze the performance of our proposed model in a simulated intensity modulated direction detection (IM/DD) optical channel, which can be found in datacenter interconnects.

The rest of the paper is structured as follows. In section II, we first introduce the equalization problem as a VAE with linear channel assumptions[20], [21]. Then we propose our innovation with a second order Volterra channel model and derive the ELBO in section II-B. In section III, we present the results of our two numerical simulation studies; first in section III-A the results from the Wiener-Hammerstein system and secondly in section III-B the results on IM/DD. Finally, in section IV, we summarize the paper, discuss the results and present an outlook for future research.

II. Methods

We begin the methods section by introducing the equalization problem and the notation used throughout the paper. Suppose that we have a sequence of information symbols $\boldsymbol{x} = \{x_0, x_1, ..., x_N\}$, where each x_i comes from a constellation $\mathcal{D}_M = \{A_1, ..., A_M\}$. In general, the constellation can be both real-valued or complex-valued, however, we will only look at real-valued constellations. The symbols are passed through a channel with (unknown) input-output relationship, $\psi(\cdot)$, such that we at the receiver observe a signal $\boldsymbol{y} = \psi(\boldsymbol{x})$. To remove any ISI and distortion that the channel has introduced, an equalizer, $f_\phi(\cdot)$, with parameters ϕ can be applied to recover the symbols $\boldsymbol{x}$ from $\boldsymbol{y}$ without explicit knowledge of ψ. We define the output of the equalizer to be $\hat{\boldsymbol{x}} = f_\phi(\boldsymbol{y})$. In the case that $\psi(\cdot)$ is linear and time-invariant, the equalizer can be implemented as a finite impulse response (FIR) filter, i.e.

$$f_\phi(\boldsymbol{y}) = \boldsymbol{h} * \boldsymbol{y}$$

in which $\boldsymbol{h}$ represent the FIR filter coefficients and $*$ is the convolution operator.

When $h(\cdot)$ is non-linear, non-linear equalizers need to be employed, such as Volterra series or neural networks. The Volterra series, which is of particular interest in this paper, is non-linear in the input due to a polynomial basis constructed from $\boldsymbol{y}$. For a second order Volterra model, the one step prediction $\hat{x}_n$ can be written as,

$$\hat{x}_n = \sum_{i=0}^{N_1} y_{n-i} h_i + \sum_{i=0}^{N_2} \sum_{j=0}^{N_2} y_{n-i} y_{n-j} H_{ij} \tag{1}$$

in which $\boldsymbol{h} \in \mathbb{R}^{N_1}$ is the first order Volterra kernel (an FIR filter), $\boldsymbol{H} \in \mathbb{R}^{N_2 \times N_2}$ is the second order Volterra kernel (matrix) and N_1 and N_2 are the respective lag lengths for the first and second order.

Most commonly *pilot* symbols are sent as part of the symbol sequence, such that ϕ can be updated *supervised* using a cost function, $\mathcal{L}(\cdot)$, towards the pilots and a gradient descent method. This yields the following optimization problem,

$$\arg\min_{\phi} \mathcal{L}(\hat{\boldsymbol{x}}, \boldsymbol{x})$$

In this formulation, if we choose $\mathcal{L}$ to be the squared error and calculate the gradient per sample, we arrive at the well-known least-mean square (LMS) optimization routine.

In the case where we do *not* have access to pilots, the equalization problem becomes *unsupervised*, often referred to as *blind* equalization. The cost function is now only defined over the output of the equalizer and is chosen to utilize an aprori known property of the constellation. For instance in CMA [17], $\mathcal{L}$ is constructed such that the equalizer output is encouraged to have the same constant modulus as the constellation.

A. Blind Equalization as a Variational Autoencoder

Casting the problem of blind channel equalization as a variational autoencoder (VAE)[20], [21] can be shown in the following way (see also Figure 1 for a visual representation). We observe a signal at the receiver, $\boldsymbol{y}$, from which we want to estimate the symbol sequence, $\boldsymbol{x}$. Given a likelihood function, $p_\theta(\boldsymbol{y}|\boldsymbol{x})$, in which θ is the collection of all channel parameters, then the posterior can be written using Bayes rule,

$$P(\boldsymbol{x}|\boldsymbol{y}) = \frac{p_\theta(\boldsymbol{y}|\boldsymbol{x})P(\boldsymbol{x})}{p(\boldsymbol{y})} = \frac{p_\theta(\boldsymbol{y}|\boldsymbol{x})P(\boldsymbol{x})}{\int p_\theta(\boldsymbol{y}|\boldsymbol{x})P(\boldsymbol{x})d\boldsymbol{x}} \tag{2}$$

in which $P(\boldsymbol{x})$ is the symbol prior modeled as a categorical distribution, which is either uniform or adapted using probabilistic constellation shaping (PCS) [24]. For all simulations in this paper, a uniform prior has been used.

The posterior exactly represents what is the desired output of an equalizer, i.e. what is the most likely symbol sequence given the received signal. In most practical applications, evaluating the integral in the denominator of (2), the model *evidence*, is practically infeasible as it involves integrating over all possible symbol sequences. Thus, we must resort to approximate methods, where the VAE comes into play. In the variational approximation, we seek a simpler distribution, $Q_\phi(\boldsymbol{x}|\boldsymbol{y})$ with free parameters ϕ, that is "close" to the true posterior, $p(\boldsymbol{x}|\boldsymbol{y})$, in some sense. A common choice is the Kullback-Leibler (KL) divergence, which can be written as,

$$\mathrm{KL}\left(Q_\phi(\boldsymbol{x}|\boldsymbol{y}) \parallel P(\boldsymbol{x}|\boldsymbol{y})\right) = \int Q_\phi(\boldsymbol{x}|\boldsymbol{y}) \log \frac{Q_\phi(\boldsymbol{x}|\boldsymbol{y})}{P(\boldsymbol{x}|\boldsymbol{y})} d\boldsymbol{x}. \tag{3}$$

Authorized licensed use limited to: Technical University of Denmark DTU Library. Downloaded on May 19,2025 at 08:41:32 UTC from IEEE Xplore. Restrictions apply.
© 2025 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2025.3570452

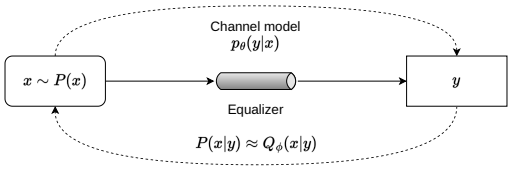


Fig. 1: Variational Autoencoder (VAE) framework for blind channel equalization. In the above $\boldsymbol{x}$ is the (unknown) symbol sequence drawn from the prior $P(\boldsymbol{x})$ and $\boldsymbol{y}$ is observed sequence at the receiver. The VAE then attempts to find an approximate posterior, $P(\boldsymbol{x}|\boldsymbol{y}) \approx Q_\phi(\boldsymbol{x}|\boldsymbol{y})$, by learning the channel parameters, θ, and equalizer, ϕ, jointly.

Expanding (3), inserting the posterior from (2) and rearranging the terms, can yield the following relation

$$\log p(\boldsymbol{y}) \geq \int \log\left[p_\theta(\boldsymbol{y}|\boldsymbol{x})\right] Q_\phi(\boldsymbol{x}|\boldsymbol{y}) d\boldsymbol{x} - \mathrm{KL}\left(Q_\phi(\boldsymbol{x}|\boldsymbol{y}) \parallel P(\boldsymbol{x})\right), \quad (4)$$

which is known as the evidence lower bound (ELBO). Maximizing the ELBO leads to a minimization of the KL-divergence in (3).

Choosing a Gaussian likelihood with isotropic noise and furthermore assuming a finite impulse repsonse (FIR) filter to model the channel, we arrive at the real-valued version of the VAE derived in [20], [21], which we will show now. We denote the time-lagged vector $\boldsymbol{x}_n = (x_n, x_{n-1}, ..., x_{n-L})$, containing all the L time-points needed to produce the channel output, y_n. The log-likelihood function in that case can be written as,

$$\log p_\theta(\boldsymbol{y}|\boldsymbol{x}) = -\frac{N}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{n=1}^{N}(y_n - \boldsymbol{x}_n^T\boldsymbol{h})^2, \quad (5)$$

in which $\boldsymbol{h}$ is the FIR filter modeling the channel response and σ^2 is the noise variance, both of which are learnable parameters in the VAE framework, i.e. $\theta = \{\boldsymbol{h}, \sigma^2\}$.

We now need to specify the approximate posterior, the Q-distribution. Following a very common practice in the variational Bayesian literature, namely that the Q-distribution factorizes, we can write it as,

$$Q_\phi(\boldsymbol{x}|\boldsymbol{y}) = \prod_n Q_\phi(x_n|\boldsymbol{y}) \quad (6)$$

in which ϕ is the collection of all parameters of the Q-distribution and the distribution $Q_\phi(x_n|\boldsymbol{y})$ is the discrete probability distribution over the constellation for the n'th symbol in the sequence. Given an equalizer output, $\hat{x}_n$, we define the individual probabilities per symbol as,

$$Q_\phi(x_n = A_m|\boldsymbol{y}) = \frac{e^{\tilde{f}_{m,n}}}{\sum_{m'} e^{\tilde{f}_{m',n}}} \quad (7)$$

$$\tilde{f}_{m,n} = \frac{-(\hat{x}_n - A_m)^2}{\sigma^2} \quad (8)$$

The interpretation of the above is that the output of the equalizer is evaluated under a (non-normalized) Gaussian density function with the constellation points as the mean and noise variance from the likelihood term. The density values are then normalized with softmax to yield a probability distribution. For equiprobable constellation points this is equivalent to the soft demapping from [21]. For all simulations in this paper, we have used a second order Volterra equalizer to model $\hat{x}$, as given in (1).

The first term in the ELBO (4), the expectation of the log-likelihood with respect to the Q-distribution, now becomes [20], [21],

$$\mathbb{E}_Q\left[\log p_\theta(\boldsymbol{y}|\boldsymbol{x})\right] = -\frac{N}{2}\log(2\pi\sigma^2) - \frac{1}{2\sigma^2}\underbrace{\mathbb{E}_Q\left[\sum_{n=1}^{N}(y_n - \boldsymbol{x}_n^T\boldsymbol{h})^2\right]}_{C} \quad (9)$$

in which $\mathbb{E}_Q[\cdot]$ is the expectation operator wrt. to the Q-distribution, i.e. $\mathbb{E}_Q\left[f(\boldsymbol{x})\right] = \int f(\boldsymbol{x})Q(\boldsymbol{x}|\boldsymbol{y})d\boldsymbol{x}$. The expectation involves integrating over all possible symbol sequences $\boldsymbol{x}$, which even for short sequence lengths is intractable and thus the above needs to be simplified.

Analyzing a single element of the sum inside C in (9), denoted c_n, one can show that this is equivalent to,

$$\begin{aligned} c_n =& y_n^2 - 2y_n\mathbb{E}_Q[\boldsymbol{x}_n]^\top\boldsymbol{h} + \mathbb{E}_Q[(\boldsymbol{x}_n^\top\boldsymbol{h})^2] \\ =& y_n^2 - 2y_n\mathbb{E}_Q[\boldsymbol{x}_n]^\top\boldsymbol{h} + (\mathbb{E}_Q[\boldsymbol{x}_n]^\top\boldsymbol{h})^2 \\ &+ (\mathbb{E}_Q[\boldsymbol{x}_n^2] - \mathbb{E}_Q[\boldsymbol{x}_n]^2])^\top\boldsymbol{h}^2 \end{aligned} \quad (10)$$

in which the expectations $\mathbb{E}_Q[\boldsymbol{x}_n]$ and $\mathbb{E}_Q[\boldsymbol{x}_n^2]$, due to the structure of the Q-distribution (6), are calculated independently per time-point. A single element, x_i, has expectations,

$$\mathbb{E}_Q\left[x_i\right] = \sum_{m=1}^{M} f_\phi(x_i = A_m|\boldsymbol{y})A_m \quad (11)$$

$$\mathbb{E}_Q\left[x_i^2\right] = \sum_{m=1}^{M} f_\phi(x_i = A_m|\boldsymbol{y})A_m^2. \quad (12)$$

The entire ELBO is differentiable wrt. θ and ϕ, and if we multiply the ELBO with -1, can thus be optimized using stochastic gradient descent, *regardless* of the equalizer parametrization (as long as the equalizer is differentiable). To estimate the noise variance, σ^2, we use the plug-in trick from [20], [21], which is achieved by analytically differentiating the ELBO wrt. to σ^2, equating to zero and solving for σ^2. This yields the solution $\sigma^2 = C/N$, which is applied in each iteration before the gradient update of the remaining parameters. Inserting the expression for σ^2 into the negative ELBO yields the loss function,

$$\mathcal{L}(\theta, \phi, \boldsymbol{y}) = \mathrm{KL}\left(Q_\phi(\boldsymbol{x}|\boldsymbol{y}) \parallel P(\boldsymbol{x})\right) + N\log C \quad (13)$$

The KL-divergence term in (13) can be calculated as,

$$\mathrm{KL}\left(Q_\phi(\boldsymbol{x}|\boldsymbol{y}) \parallel P(\boldsymbol{x})\right) = \sum_{n=1}^{N}\sum_{m=1}^{M} Q_\phi(x_n = A_m|\boldsymbol{y})\log\frac{Q_\phi(x_n = A_m|\boldsymbol{y})}{P(x_n = A_m)} \quad (14)$$

which in the case of a uniform prior over symbols simplifies to the negative entropy of the Q-distribution. The VAE is fitted using a stochastic gradient descent algorithm utilizing

Authorized licensed use limited to: Technical University of Denmark DTU Library. Downloaded on May 19,2025 at 08:41:32 UTC from IEEE Xplore. Restrictions apply.
© 2025 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2025.3570452

Algorithm 1 Fitting a Variational Autoencoder (VAE) with Stochastic Gradient Descent

Input: Data at receiver $\boldsymbol{y}$, learning rate η, batch size B
Output: Trained equalizer and channel parameters ϕ and θ
Initialize equalizer parameters ϕ and channel parameters θ, compute total number of batches, N_{batch}
for b $= 1$ to N_{batch} **do**
    Compute equalizer signal $\hat{\boldsymbol{x}}^{(b)}$, for batch $\boldsymbol{y}^{(b)}$
    Compute ELBO $\mathcal{L}$ with (13), including term C
    Compute noise variance $\sigma^2 = C/B$
    Compute gradients $\nabla_\phi \mathcal{L}$ and $\nabla_\theta \mathcal{L}$
    Update parameters:
        $\phi \leftarrow \phi + \eta \Delta_b(\nabla_\phi \mathcal{L})$, $\theta \leftarrow \theta + \eta \Delta_b(\nabla_\theta \mathcal{L})$
        where Δ_b is the Adam [25] update
end for

the Adam optimizer [25]. An overview of the update scheme can be seen in Algorithm 1.

It should be noted that the VAE framework can be used with an arbitrary non-linear channel model. However, in that case, we can no longer calculate the gradient of the loss function analytically and must resort to approximate methods. It was shown in [20], that this can be done utilizing the Gumbel-Softmax approximation [26]. This approach involves sampling the gradients, which we would expect yields more noise in optimization and has not been explored in this paper.

Commonly, equalization is performed in an oversampled domain with multiple samples per symbol (sps) leading to $\boldsymbol{y}$ and $\mathbb{E}_Q[\boldsymbol{x}]$ not having equal length. In this case the expectation vectors are upsampled by inserting zeros between the symbols to match the length of $\boldsymbol{y}$ as suggested in [21].

B. Variational Autoencoder with Second Order Volterra Channel Model

We now turn to the case, where the channel model is assumed to have a second order Volterra series structure, and we derive an analytical expression for the expected log-likelihood. Given the time-lagged vector $\boldsymbol{x}_n$, then the second order Volterra model can be written as

$$\hat{y}_n = \boldsymbol{x}_n^\top \boldsymbol{h} + \boldsymbol{x}_n^\top \boldsymbol{H} \boldsymbol{x}_n, \tag{15}$$

in which $\boldsymbol{h}$ and $\boldsymbol{H}$ are the first and second order Volterra kernels, respectively. The matrix $\boldsymbol{H}$ is symmetric, i.e. $H_{ij} = H_{ji}$. We have for simplicity assumed here that both the first and second order Volterra terms use the same input vector $\boldsymbol{x}_n$.

As in (5) for the FIR channel model, we are interested in deriving the analytical expression for the expected log-likelihood of the VAE, more specifically the subterm C. We note that it is necessary to obtain closed-form solutions of the expectations to make the calculation of the ELBO practically feasible. Otherwise, we would have to either— integrate over all possible symbol sequences, which even for short sequence lengths is computationally impractical or resort to sampling, which has high variance and would slow down convergence speed.

Replacing the first order model with a second order one, we arrive at

$$C = \mathbb{E}_Q \left[\sum_{n=1}^{N} \left(y_n - \left(\boldsymbol{x}_n^\top \boldsymbol{h} + \boldsymbol{x}_n^\top \boldsymbol{H} \boldsymbol{x}_n \right) \right)^2 \right] \tag{16}$$

Analyzing a single element of the sum in (16), we arrive at

$$\begin{aligned} c_n =& \mathbb{E}_Q \left[(y_n - \hat{y}_n)^2 \right] \\ =& \mathbb{E}_Q \left[\left(y_n - (\boldsymbol{x}_n^\top \boldsymbol{h} + \boldsymbol{x}_n^\top \boldsymbol{H} \boldsymbol{x}_n) \right)^2 \right] \\ =& y_n^2 - 2 y_n \mathbb{E}_Q[\boldsymbol{x}_n]^\top \boldsymbol{h} + \mathbb{E}_Q[(\boldsymbol{x}_n^\top \boldsymbol{h})^2] \\ & - 2 y_n \mathbb{E}_Q[\boldsymbol{x}_n^\top \boldsymbol{H} \boldsymbol{x}_n] + 2 \mathbb{E}_Q[\boldsymbol{x}_n^\top \boldsymbol{h} \boldsymbol{x}_n^\top \boldsymbol{H} \boldsymbol{x}_n] \\ & + \mathbb{E}_Q[(\boldsymbol{x}_n^\top \boldsymbol{H} \boldsymbol{x}_n)^2] \end{aligned} \tag{17}$$

The terms $\mathbb{E}_Q[\boldsymbol{x}_n]^\top \boldsymbol{h}$ and $\mathbb{E}_Q[(\boldsymbol{x}_n^\top \boldsymbol{h})^2]$ are identical to terms found in (10). In the following we will, look at each of the last three terms and derive how they can be calculated analytically, given the moments of x_n assumed to follow our Q distribution. The derivation of the three terms follows the same structure, namely to write out the expectations as summations, identify the matching indices such that the expectation can be simplified ($\mathbb{E}[x_i x_i] = \mathbb{E}[x_i^2]$) and appropriately subtracting terms that arise from doing full summations. For a more detailed derivation, including all terms from (17), we refer the reader to the supplementary material.

In the following, we will use a simplified notation where we suppress the subscript Q in the expectation, the index n is removed and indices i, j, k and ℓ are implicitly summed over, e.g. $\mathbb{E}_Q[\boldsymbol{x}_n^\top \boldsymbol{h}] = \mathbb{E}[x_i h_i]$. Using this notation we arrive at,

$$\begin{aligned} \mathbb{E}_Q[\boldsymbol{x}_n^\top \boldsymbol{H} \boldsymbol{x}_n] &= \mathbb{E}[x_i H_{ij} x_j] \\ &= \mathbb{E}[x_i]\mathbb{E}[x_j] H_{ij} + \mathbb{E}[x_i^2] H_{ii} - \mathbb{E}[x_i]^2 H_{ii} \end{aligned} \tag{18}$$

The cross-term between first and second order kernel becomes,

$$\begin{aligned} \mathbb{E}_Q[\boldsymbol{x}_n^\top \boldsymbol{h} \boldsymbol{x}_n^\top \boldsymbol{H} \boldsymbol{x}_n] &= \mathbb{E}[x_i x_j x_k h_i H_{jk}] \\ &= \mathbb{E}[x_i]\mathbb{E}[x_j]\mathbb{E}[x_k] h_i H_{jk} \\ &\quad + \left(\mathbb{E}[x_i^2]\mathbb{E}[x_j] - \mathbb{E}[x_i]^2 \mathbb{E}[x_j] \right) (2 h_i H_{ij} + h_j H_{ii}) \\ &\quad + \left(\mathbb{E}[x_i^3] - 3\mathbb{E}[x_i^2]\mathbb{E}[x_i] + 2\mathbb{E}[x_i]^3 \right) h_i H_{ii} \end{aligned} \tag{19}$$

The squared second order kernel term becomes,

$$\begin{aligned} &\mathbb{E}_Q[(\boldsymbol{x}_n^\top \boldsymbol{H} \boldsymbol{x}_n)^2] = \mathbb{E}[x_i x_j x_k x_\ell H_{ij} H_{k\ell}] \\ &= \mathbb{E}[x_i]\mathbb{E}[x_j]\mathbb{E}[x_k]\mathbb{E}[x_\ell] H_{ij} H_{k\ell} \\ &\quad + (\mathbb{E}[x_i^2]\mathbb{E}[x_j]\mathbb{E}[x_k] - \mathbb{E}[x_i]^2 \mathbb{E}[x_j]\mathbb{E}[x_k]) \cdot \\ &\qquad (2 H_{ii} H_{jk} + 4 H_{ij} H_{ik}) \\ &\quad + (\mathbb{E}[x_i^2]\mathbb{E}[x_j^2] - \mathbb{E}[x_i^2]\mathbb{E}[x_j]^2 - \mathbb{E}[x_i]^2 \mathbb{E}[x_j^2] + \mathbb{E}[x_i]^2 \mathbb{E}[x_j]^2) \cdot \\ &\qquad (2 H_{ij} H_{ij} + H_{ii} H_{jj}) \\ &\quad + (\mathbb{E}[x_i^3]\mathbb{E}[x_j] - 3\mathbb{E}[x_i^2]\mathbb{E}[x_i]\mathbb{E}[x_j] + 2\mathbb{E}[x_i]^3 \mathbb{E}[x_j]) \cdot \\ &\qquad (4 H_{ii} H_{ij}) \\ &\quad + \big(\mathbb{E}[x_i^4] + 12\mathbb{E}[x_i^2]\mathbb{E}[x_i]^2 - 3\mathbb{E}[x_i^2]^2 \\ &\qquad - 4\mathbb{E}[x_i^3]\mathbb{E}[x_i] - 6\mathbb{E}[x_i]^4 \big) H_{ii}^2 \end{aligned} \tag{20}$$

The higher order moments ($\mathbb{E}[x^3]$ and $\mathbb{E}[x^4]$) can be calculated similarly to (11) and (12). The resulting loss function

Authorized licensed use limited to: Technical University of Denmark DTU Library. Downloaded on May 19,2025 at 08:41:32 UTC from IEEE Xplore. Restrictions apply.
© 2025 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2025.3570452

has the same structure as the VAE from the previous section, cf. (13), with the newly derived term C (16) inserted.

We implemented the model and loss function using PyTorch[1] to allow for easy-to-use automatic differentiation and optimization routines. Our implementation follows the structure from [21][2] and the same general update rules as in Algorithm 1.

III. Numerical Results

We present numerical results from two different simulation models, a Wiener-Hammerstein channel and an intensity modulated direct detection (IM/DD) optical communication system. In the results, we compare five equalization schemes,

VAE Variational autoencoder from [20], [21] with a second order Volterra equalizer.

V2VAE Variational autoencoder with second order Volterra channel model (this work) and a second order Volterra equalizer.

MP-VAE Variational autoencoder with a memory polynomial channel model, first implemented as a baseline model in [23]. We use a second order Volterra equalizer.

Volterra Standard second order Volterra series equalizer with pilots (non-blind).

FFE Standard linear feed-forward equalizer with pilots (non-blind).

CNN A supervised convolutional neural network (non-blind).

We stress that both the VAE and the V2VAE are capable of doing (blind) non-linear equalization, but it is in their channel assumptions that they differ. In all simulations, we used the Adam optimizer [25] and screened the learning rate between 5 values spaced in the range $(5 \cdot 10^{-5}, 5 \cdot 10^{-3})$. Furthermore, the batch size was exhaustively sweeped together with the learning rate for the values $[500, 1000, 2000]$. In the following, the reported symbol error rate (SER) values are for the best performing combination of learning rate and batch size. We used a step-wise learning rate scheduling, where, given a number of iterations, N_{iter}, the learning rate was reduced every $N_{\text{iter}}/10$ iteration, such that the final learning rate was 100 times lower than the initial value. We use $2 \cdot 10^6$ symbols for training and 10^6 symbols for SER calculation after convergence. The supervised methods, that use pilots, are trained using the mean square error (MSE) cost function averaged over a batch of symbols. For a batch size of 1 this would be equivalent to the least mean square (LMS) algorithm. In all simulations, we used $N_{\text{taps}}^{(1)} = 25$ taps in the FIR part of the equalizer (all methods except CNN), $N_{\text{taps}}^{(2)} = 15$ taps in the second order equalizer kernel (VAE, MP-VAE, V2VAE and Volterra) and a channel memory $N_{\text{channel}} = 25$ (VAE, MP-VAE and V2VAE). The V2VAE uses the same number of lags in both first and second order to model the channel, meaning that the symmetric second order kernel has on the order of $O(N_{\text{channel}}^2)$ free parameters. A VAE with a simplified memory polynomial (MP) as channel model was used as a baseline model in [23], which has an analytical ELBO. To be comparable to the V2VAE, we implemented a second order version of the MP-VAE, yielding $2N_{\text{channel}}$ parameters in the channel model. For more details and a derivation of the ELBO see the appendix. This formulation of the memory polynomial can be interpreted as a second order Volterra model, where the second order kernel only has non-zero elements in the diagonal.

To give an estimate of the achievable SER in a non-linear channel, we furthermore fit a supervised convolutional neural network (CNN). The CNN used consists of $N_{\text{filters}} = 20$ convolutional kernels of lengths $L = 55$ with a stride of sps. The output of each filter is stacked, passed to a batch normalization layer [27] and mapped through a fully-connected feed-forward neural network with $N_{\text{layers}} = 5$ with rectified linear unit (ReLU) activations [28]. After the last layer a linear layer is applied with a single output dimension to form the symbol decision, $\hat{x}$. The CNN is trained with the same cost-function (MSE) as the other supervised methods.

A. Wiener-Hammerstein Channel

The Wiener-Hammerstein system is a well-studied general function [29], used to model non-linear dynamic systems such as loudspeakers in acoustic echo-cancellation systems [30], power amplifiers in radio communication [31] and transmitters in optical communication [32], to mention a few. The Wiener-Hammerstein system is comprised of two finite impulse response (FIR) filters, with a memory-less non-linearity, $g(\cdot)$, in-between. We choose g to be a second order polynomial and the system transfer function, $\psi_{\text{wh}}(x)$, can be written as,

$$\begin{aligned} \psi_{\text{wh}}(x) &= h_2 * (g\,(h_1 * x))\,, \\ g(x) &= (1-\alpha)x + \alpha x^2. \end{aligned} \tag{21}$$

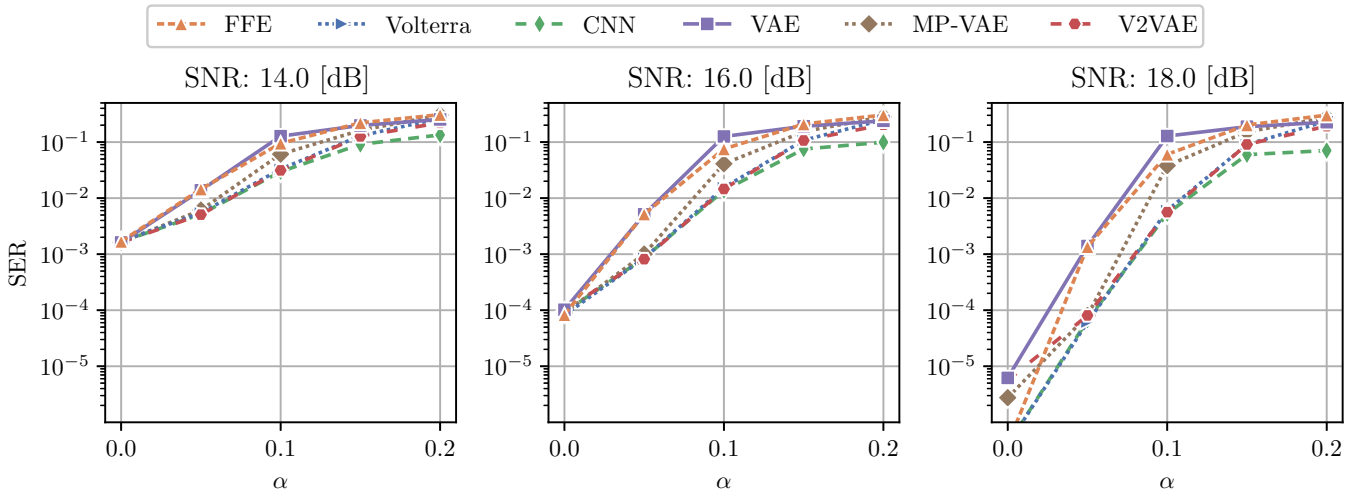
For this choice of non-linearity, the Wiener-Hammerstein system is a subclass of the second order Volterra system. The Wiener-Hammerstein system is inserted into a simple additive white Gaussian noise (AWGN) communication channel. We generate a random sequence of symbols drawn from the constellation $\mathcal{A} = \{-3, -1, 1, 3\}$, also known as pulse-amplitude modulation with order 4 (PAM-4). The symbols are then up-sampled by an oversampling factor of 4 after which they are pulse-shaped with a root-raised cosine (RRC) filter with rolloff $\rho = 0.1$. The signal is then passed to the Wiener-Hammerstein system from (21). The FIR filters have coefficients, $h_1 = [1.0, 0.3, 0.1]$ and $h_2 = [1.0, -0.2, 0.02]$ (designed to each be minimum-phase and have two zeros) and the non-linearity α is varied during the simulations. Both h_1 and h_2 are upsampled (zero-insertion and interpolation) with the same oversampling factor as the symbols before they are applied in the channel. After the Wiener-Hammerstein system, AWGN is added to yield a pre-specified signal-to-noise ratio (SNR) of $\frac{\mathbb{E}_s[\psi_{\text{wh}}(x)^2]}{\sigma^2}$, where $\mathbb{E}_s$ is the empirical average energy-per-symbol and σ^2 is the noise variance. Matched filtering (RRC) is applied, the sequence is decimated to 2 samples per symbol and the resulting signal is synchronized with the symbol sequence. The different equalizers are then

[1]Our code is available from: https://github.com/sfvnielsen/volterra-vae

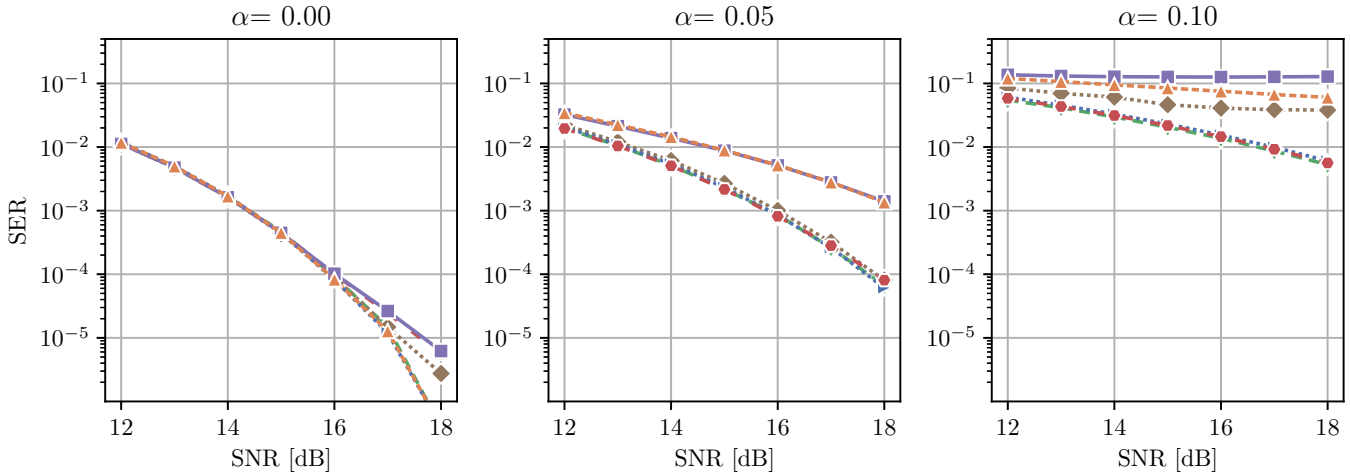
[2]We would like to give kudos to the authors from [21] for putting their code on Github.

Authorized licensed use limited to: Technical University of Denmark DTU Library. Downloaded on May 19,2025 at 08:41:32 UTC from IEEE Xplore. Restrictions apply.
© 2025 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2025.3570452



(a) SER as a function of non-linear coefficient α.



(b) SER as a function of SNR.

Fig. 2: Symbol error-rate (SER) results for Wiener-Hammerstein channel with varying degree of non-linearity, α in panel 2a and varying SNR in panel 2b. Each simulation was restarted 5 times, thus the curves represent the average over runs.

fitted to the resulting sequence. After convergence, a new test set is generated with the same steps as above and the symbol error rate (SER) is calculated. For the two supervised methods (FFE and Volterra), we map the output of the equalizer to the nearest constellation point in the standard Euclidean sense and count the errors. For the two variational auto-encoders (VAE and V2VAE), we use the estimated symbol probabilities from the Q-distribution, pick the most likely symbol under the model and then count the errors.

The results of varying the SNR and the non-linearity coefficient α can be seen in Figure 2. In the linear regime ($\alpha = 0$), we see that all methods generally achieve similar SER, i.e. the non-linear methods can adapt their non-linear components to the problem at hand, with an exception in the high SNR case where blind methods incur a small penalty. As we increase the quadratic term ($\alpha > 0$), we see that, unsurprisingly, the SER of the linear FFE starts to increase more than than the supervised Volterra method. The *unsupervised* V2VAE closely follows the performance of its supervised counterpart, whereas the VAE with mismatched channel assumptions, follows more the trend of the linear FFE. The MP-VAE, which has a slighty more advanced channel model than the standard VAE, performs almost on par with the V2VAE except in the strong non-linear regime ($\alpha = 0.1$), where the more advanced channel model of V2VAE has an advantage. The supervised CNN, with by far the most parameters and modeling capacity is (unsuprisingly) best across the board. However, in most cases the Volterra methods (V2VAE, MP-VAE and Volterra) rival the performance of the over-parameterized CNN.

To investigate the unsupervised methods (VAE, MP-VAE and V2VAE) convergence properties and ability to track changes in the system, we devised a simple change-point test. We use the Wiener-Hammerstein system described above with $\alpha = 0.05$, but change the first FIR filter, h_1, to a new set of coefficients, $h_1^* = [1.0, 0.5, 0.1525]$, after $2.5 \cdot 10^6$ symbols and continue to alternate between h_1 and h_1^* every $2.5 \cdot 10^6$

Authorized licensed use limited to: Technical University of Denmark DTU Library. Downloaded on May 19,2025 at 08:41:32 UTC from IEEE Xplore. Restrictions apply.
© 2025 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2025.3570452

symbols. During this simulation we disable the learning rate scheduling and run with a fixed learning rate. We report the loss function value as a function of processed batches and the average SER on a held-out validation set (10^6 symbols) grouped by h_1 and h_1^* can be seen in Figure 3.

In general, the V2VAE converges to a lower loss function value compared to the VAE in this non-linear channel, where the VAEs channel model assumptions are violated. The MP-VAE converges to a loss function value between the VAE and V2VAE. As expected, the lower loss function value also translates to better SER performance, seen in the right panel of Figure 3. However, we also note that the V2VAE converges slightly slower to a stable loss-function level than the VAE. We attribute this to the more complex channel model that the V2VAE has to fit.

B. Intensity Modulated Direct Detection System

In this section, we study a simulated optical communication system based on unamplified intensity modulated direct detection (IM/DD), commonly found in datacenter interconnects [33], inspired by the system model in [34]. We again use the PAM-4 modulation format with constellation $\mathcal{A} = \{-3, -1, 1, 3\}$. We use the same transmitter processing as in section III-A (up-sampling by 4 and pulse-shaping with RRC and rolloff $\rho = 0.1$) with a baud rate $R_s = 100$ GBaud. The signal is then passed to a digital-to-analog converter (DAC), comprising of a voltage normalization step to the range $[-\frac{1}{2}, \frac{1}{2}]$, multiplication with a peak-to-peak voltage, V_{pp} and application of a 5th order Bessel low-pass filter. The 3-db cutoff frequency of the low-pass filter was set to 55 GHz. The voltage signal, $V(t)$, is then used as input to a Mach-Zehnder modulator (MZM), which yields the optical signal,

$$E(t) = \sqrt{P_{in}} \cos\left(\frac{1}{2V_\pi}(V(t) + V_b)\right), \tag{22}$$

in which $P_{in} = -3.0$ dBm is the power of the laser at the input of the modulator and $V_\pi = 2\,\mathrm{V}$ and $V_b = -0.5\,\mathrm{V}$ are parameters of the MZM. A plot of the modulator characteristic and eyediagrams in the noiseless case can be seen in Figure 4. A standard single mode fiber model [35] is used to model chromatic dispersion and fiber loss. The fiber has a dispersion slope $S_0 = 0.092$ ps/(nm^2 km), zero-dispersion wavelength $\lambda_0 = 1310$ nm, attenuation $\alpha_{smf} = 0.2$ dB/km. We use a laser wavelength of $\lambda = 1270$ nm, which yields a dispersion parameter of $D = \frac{S_0}{4}\left(\lambda - \frac{\lambda_0^4}{\lambda^3}\right) \approx -15.43$ ps/(nm $\cdot$ km). At the receiver, the signal is converted to voltage domain using a square-law detector with thermal and shot noise modeled as AWGN. The noise variances are parameterized as,

$$\sigma_t^2 = \frac{4 \cdot k \cdot T \cdot F_s}{B \cdot Z} \tag{23}$$

$$\sigma_s^2 = \frac{2e_c\left(R \cdot \mathbb{E}[|y|^2] + I_d\right) F_s}{B}, \tag{24}$$

in which k is the Boltzmann constant, $T = 293\,\mathrm{K}$ is the temperature of the photodiode, F_s is the sampling frequency, $B = 55$ GHz is the assumed bandwidth of the photodiode, $Z = 50\Omega$ is the impedance load, e_c is the electron charge, $R = 1$ [A/W] is the responsivity of the photodiode, $\mathbb{E}[|y|^2]$ is the empirical average power received and $I_d = 1{\cdot}10^{-8}$ A is the dark current. The signal is converted to digital domain again using an analog-to-digital converter (ADC) with the same bandwidth limitation and filters as the DAC. The matched filter (RRC) is applied in the digital domain, and finally the signal is down-sampled to 2 samples-per-symbol and the symbol sequence is synchronized to the received signal. The equalizers are then fitted to the training sequence and the SER is calculated in the same way as for the Wiener-Hammerstein system on a new test set. We note that the laser power is assumed to be fixed and we do not use an amplifier in this system. Thus the only way to increase the effective "SNR" is by increasing V_{pp}. However, at some point, defined by the modulator characteristic, the modulator will enter a regime where non-linear distortion will start to hamper the SER.

The SER results for varying V_{pp} and the fiber length can be seen in Figure 5. In the back-to-back (B2B) condition (fiber length of 0 km), the main sources of distortion is the ISI introduced by the bandwidth limitation in the DAC and ADC and the non-linearity in the modulator as the V_{pp} is increased. In the low voltage regime ($V_{pp} < 0.8\,\mathrm{V}$), there is generally little to no difference in performance across the methods, as the modulator is operating in the linear range. As V_{pp} increases the non-linear methods start to gain an advantage over the supervised linear FFE. The supervised Volterra and CNN methods are generally performing best, closely followed by the unsupervised V2VAE and MP-VAE. When V_{pp} reaches the highly non-linear regime, we see the biggest advantage of having a non-linear channel assumption showcased by the V2VAE performing significantly better than the standard VAE, even though they have the same equalizer parametrization. Similarly, the MP-VAE achieves a lower SER than the VAE due to its more advanced channel model. Comparing the V2VAE and the MP-VAE there seems to be an advantage of using the full second order kernel in the channel model (V2VAE), most noticeably in the highly non-linear regime ($V_{pp} > 1.0\,\mathrm{V}$). Looking at longer fiber lengths (1 and 2 km), the chromatic dispersion becomes the main source of distortion and the advantage of using non-linear methods lessens. As for the Wiener-Hammerstein channel, the CNN provides an empirical estimate for the achievable SER in this system. In the B2B condition, the gap to the CNN is almost closed by the V2VAE and Volterra methods. However, the gap in the non-linear region of the modulator ($V_{pp} > 1.0\,\mathrm{V}$) increases as we increase the fiber length.

We also ran the simulations with a electro-absorption modulator (EAM) [36], the result of which can be seen in the supplementary material.

IV. Discussion and Conclusion

We investigated the impact of channel modeling assumptions in a VAE equalizer framework. We extended the channel model (decoder) to be non-linear with a specific structure, namely a second order Volterra series and derived the analytical ELBO for optimizing the equalizer. In both simulation studies, Wiener-Hammerstein and IM/DD, we found support

Authorized licensed use limited to: Technical University of Denmark DTU Library. Downloaded on May 19,2025 at 08:41:32 UTC from IEEE Xplore. Restrictions apply.
© 2025 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2025.3570452

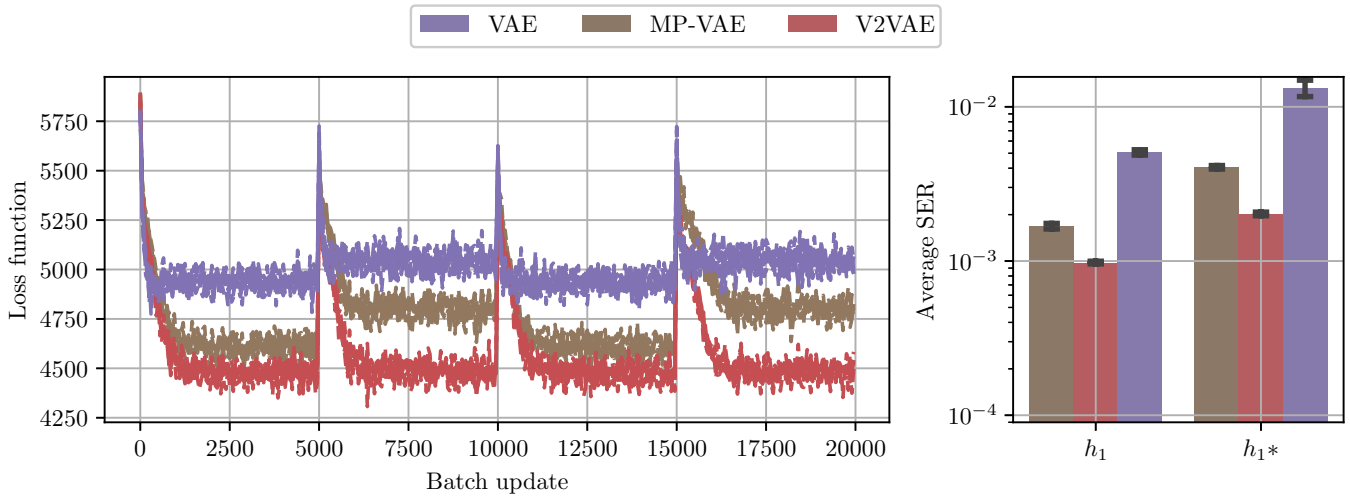
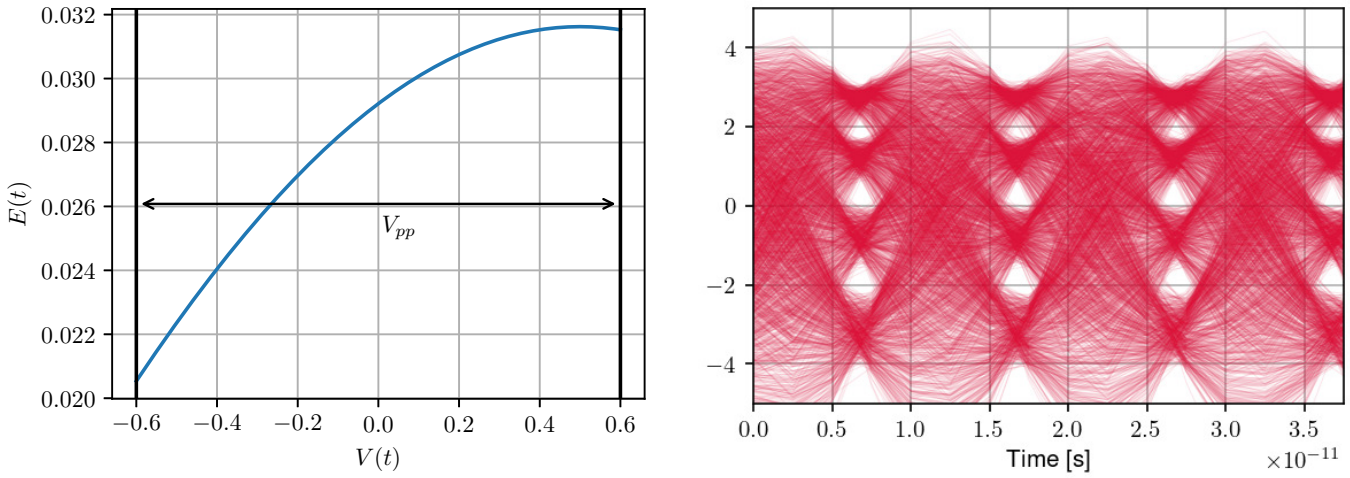


Fig. 3: Tracking in Wiener-Hammerstein channel ($\alpha = 0.05$ and SNR = 16 dB). We changed the coefficients of the first FIR filter in the system every $5 \cdot 10^6$ symbols, alternating between two sets of coefficients, h_1 and h_1*. The test was restarted 5 times with a new seed. We show the loss function (left panel), for all restarts, as a function of batch update (batch size of 500 symbols) and the average SER (right panel) per system with errorbars indicating confidence interval estimated over seeds. The SER was calculated on an independent validation set of size 10^6 symbols.



(a) Voltage-to-optical characteristic of the modulator.

(b) Eyediagram after matched filter (noiseless case).

Fig. 4: Mach-Zehnder modulator (MZM) and a accompanying eyediagram after the receiver filter for $V_{pp} = 1.2\,\mathrm{V}$, $V_\pi = 2\,\mathrm{V}$ and $V_b = -0.5\,\mathrm{V}$. No noise was added in the photodiode, such that the eyediagram only shows the impact of the non-linearity.

that appropriate channel modeling leads to better equalization performance in the unsupervised VAE framework. Looking at the models ability to track changes in the system, we demonstrated that V2VAE can achieve a lower cost function value compared to the VAE, but does so at a slower pace, requiring more symbols for convergence. As discussed and proposed in [21], given enough computational resources, one could improve the convergence time by having overlapping batches and thus effectively doing more gradient updates per symbols (denoted the *flex*-scheme in [21]). Another aspect of this is the memory efficiency, where the standard algorithms like FFE updated with LMS, can be updated once per symbol and only needs to store the delayline to do the gradient update. In VAE models, batched updating is preferred to reduce the variance in the stochastic gradient calculation. However, this adds a higher memory requirement in practical systems, which in turn will drive batch sizes to be as small as possible. In this paper, the minimum batch size that still yields good convergence behaviour was not explored, but could be an interesting avenue of future research.

We note that the improvement in modeling capabilities by the V2VAE does not come for free. The calculation of the cost function and the associated gradients are more expensive to compute, compared to the linear VAE and the MP-VAE from [23]. For the VAE and the MP-VAE, the calculation of the cost function for one time-point scales with $O(N_{\text{channel}})$, as

Authorized licensed use limited to: Technical University of Denmark DTU Library. Downloaded on May 19,2025 at 08:41:32 UTC from IEEE Xplore. Restrictions apply.
© 2025 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2025.3570452

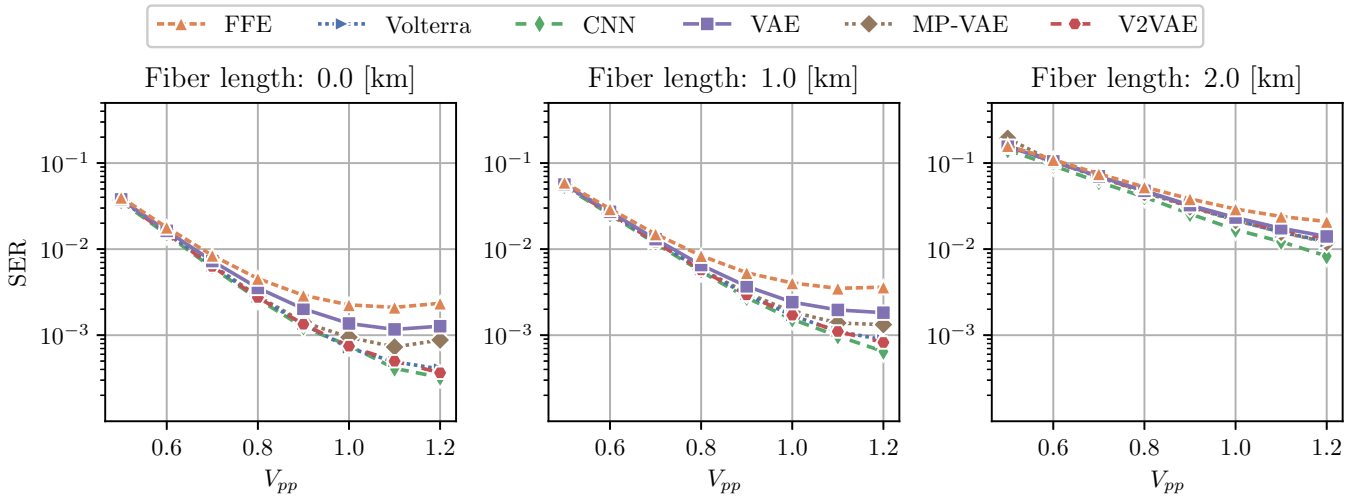


Fig. 5: Results for IM/DD channel with varying fiber length and varying DAC peak-to-peak voltage, V_{pp} [V]. Each simulation was restarted 5 times with different random seeds and the resulting curves represent an average over runs.

all the terms can be written as convolutions and dot products. However, the V2VAE scales with $O(N^2_{\text{channel}})$, when taking into account the optimal order of summation, due to the squared second order kernel term in (20). We hypothesize that a more efficient approximate algorithm for the V2VAE could be derived, by decimating the calculation of the more expensive terms in the ELBO and low-pass filtering. This would reduce how often the $O(N^2_{\text{channel}})$ terms needed to be calculated, however, the quality of that approximation is yet to be studied.

The authors in [20] used the VAE in a low-density parity-check (LDPC) coded data transmission scenario, where the estimated symbol probabilities were used in conjunction with a belief propagation algorithm to decode the most probable bit sequence. The V2VAE also admits to this extension, and could potentially lead to better decoding performance in non-linear channels due to the more flexible channel modeling.

A natural extension, would be to derive the model for complex-valued constellations, as done originally in both [20] and [21]. This would allow the V2VAE to be applied to coherent optical transmission and model cross-talk between different modes in the fiber as in [21]. However, that derivation has been deemed out of scope for this paper.

We used a second order Volterra model as the channel model, both due to the Volterra models popluarity for non-linear system identification and non-linear equalization. However, future work could explore other structures for non-linear channel modeling while keeping the ELBO analytical, that might computationally scale better in the channel memory.

REFERENCES

[1] D. Mytton and M. Ashtine, "Sources of data center energy estimates: A comprehensive review," *Joule*, vol. 6, pp. 2032–2056, Sept. 2022.

[2] P. K. Singya, N. Kumar, and V. Bhatia, "Mitigating NLD for Wireless Networks: Effect of Nonlinear Power Amplifiers on Future Wireless Communication Networks," *IEEE Microwave Magazine*, vol. 18, pp. 73–90, July 2017.

[3] K. Ying, Z. Yu, R. J. Baxley, H. Qian, G.-K. Chang, and G. T. Zhou, "Nonlinear distortion mitigation in visible light communications," *IEEE Wireless Communications*, vol. 22, pp. 36–45, Apr. 2015.

[4] J. Li, X. Su, T. Ye, H. Nakashima, T. Hoshida, and Z. Tao, "Characterization of Nonlinear Distortion in Intensity Modulation and Direct Detection Systems," *Journal of Lightwave Technology*, vol. 41, pp. 3513–3521, June 2023.

[5] J. Proakis and M. Salehi, *Digital Communications*. McGraw Hill,, 2008.

[6] Q. Yu, "On the Decision-Feedback Equalizer in Optically Amplified Direct-Detection Systems," *Journal of Lightwave Technology*, vol. 25, pp. 2090–2097, Aug. 2007.

[7] N. Stojanovic, F. Karinou, Z. Qiang, and C. Prodaniuc, "Volterra and Wiener Equalizers for Short-Reach 100G PAM-4 Applications," *Journal of Lightwave Technology*, vol. 35, pp. 4583–4594, Nov. 2017.

[8] M. Schetzen, "Nonlinear System Modelling and Analysis from the Volterra and Wiener Perspective," in *Block-Oriented Nonlinear System Identification*, pp. 13–24, Springer, London, 2010.

[9] J. Staudinger, J.-C. Nanan, and J. Wood, "Memory fading Volterra series model for high power infrastructure amplifiers," in *2010 IEEE Radio and Wireless Symposium (RWS)*, pp. 184–187, Jan. 2010.

[10] K. J. Friston, A. Mechelli, R. Turner, and C. J. Price, "Nonlinear Responses in fMRI: The Balloon Model, Volterra Kernels, and Other Hemodynamics," *NeuroImage*, vol. 12, pp. 466–477, Oct. 2000.

[11] X. Deng, S. Mardanikorani, Y. Wu, K. Arulandu, B. Chen, A. M. Khalid, and J.-P. M. G. Linnartz, "Mitigating LED Nonlinearity to Enhance Visible Light Communications," *IEEE Transactions on Communications*, vol. 66, pp. 5593–5607, Nov. 2018.

[12] G. Gibson, S. Siu, and C. Cowan, "Application of multilayer perceptrons as adaptive channel equalisers," in *Adaptive Systems in Control and Signal Processing 1989*, IFAC Symposia Series, pp. 573–578, Oxford: Pergamon, 1990.

[13] C. You and D. Hong, "Nonlinear blind equalization schemes using complex-valued multilayer feedforward neural networks," *IEEE Transactions on Neural Networks*, vol. 9, pp. 1442–1455, Nov. 1998.

[14] J. Estaran, R. Rios-Mueller, M. A. Mestre, F. Jorge, H. Mardoyan, A. Konczykowska, J.-Y. Dupuy, and S. Bigo, "Artificial Neural Networks for Linear and Non-Linear Impairment Mitigation in High-Baudrate IM/DD Systems," in *42nd European Conference on Optical Communication (ECOC)*, pp. 1–3, Sept. 2016.

[15] L. Huang, Y. Xu, W. Jiang, L. Xue, W. Hu, and L. Yi, "Performance and complexity analysis of conventional and deep learning equalizers for the high-speed IMDD PON," *Journal of Lightwave Technology*, vol. 40, pp. 4528–4538, July 2022.

[16] Y. Sato, "Method of self-recovering equalization for multilevel amplitude-modulation systems," *Ieee Transactions on Communications*, vol. CO23, no. 6, pp. 679–682, 1975.

[17] D. Godard, "Self-Recovering Equalization and Carrier Tracking in Two-Dimensional Data Communication Systems," *IEEE Transactions on Communications*, vol. 28, pp. 1867–1875, Nov. 1980.

Authorized licensed use limited to: Technical University of Denmark DTU Library. Downloaded on May 19,2025 at 08:41:32 UTC from IEEE Xplore. Restrictions apply.
© 2025 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Cognitive Communications and Networking. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2025.3570452

[18] J. Yang, J.-J. Werner, and G. Dumont, "The multimodulus blind equalization and its generalized algorithms," *IEEE Journal on Selected Areas in Communications*, vol. 20, pp. 997–1015, June 2002.
[19] A. Caciularu and D. Burshtein, "Blind Channel Equalization Using Variational Autoencoders," in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, May 2018.
[20] A. Caciularu and D. Burshtein, "Unsupervised Linear and Nonlinear Channel Equalization and Decoding Using Variational Autoencoders," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, pp. 1003–1018, Sept. 2020.
[21] V. Lauinger, F. Buchali, and L. Schmalen, "Blind Equalization and Channel Estimation in Coherent Optical Communications Using Variational Autoencoders," *IEEE Journal on Selected Areas in Communications*, vol. 40, pp. 2529–2539, Sept. 2022.
[22] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," Dec. 2022.
[23] J. Song, V. Lauinger, Y. Wu, C. Häger, J. Schröder, A. G. i Amat, L. Schmalen, and H. Wymeersch, "Blind Channel Equalization Using Vector-Quantized Variational Autoencoders," Feb. 2023.
[24] J. Cho and P. J. Winzer, "Probabilistic Constellation Shaping for Optical Fiber Communications," *Journal of Lightwave Technology*, vol. 37, pp. 1590–1607, Mar. 2019.
[25] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.
[26] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *5th International Conference on Learning Representations (ICLR)*, 2017.
[27] R. Arora, A. Basu, P. Mianjy, and A. Mukherjee, "Understanding deep neural networks with rectified linear units," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, OpenReview.net, 2018.
[28] Y. LeCun, L. Bottou, G. B. Orr, and K. R. Müller, "Efficient BackProp," in *Neural Networks: Tricks of the Trade*, pp. 9–50, Berlin, Heidelberg: Springer, 1998.
[29] S. A. Billings and S. Y. Fakhouri, "Identification of systems containing linear dynamic and static nonlinear elements," *Automatica*, vol. 18, pp. 15–26, Jan. 1982.
[30] M. M. Halimeh, C. Huemmer, and W. Kellermann, "A Neural Network-Based Nonlinear Acoustic Echo Canceller," *IEEE Signal Processing Letters*, vol. 26, pp. 1827–1831, Dec. 2019.
[31] A. Bolstad, B. A. Miller, J. Goodman, J. Vian, and J. Kalyanam, "Identification and compensation of Wiener-Hammerstein systems with feedback," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4056–4059, May 2011.
[32] T. Sasai, M. Nakamura, E. Yamazaki, A. Matsushita, S. Okamoto, K. Horikoshi, and Y. Kisaka, "Wiener-Hammerstein model and its learning for nonlinear digital pre-distortion of optical transmitters," *Optics Express*, vol. 28, pp. 30952–30963, Oct. 2020.
[33] K. Zhong, X. Zhou, J. Huo, C. Yu, C. Lu, and A. P. T. Lau, "Digital Signal Processing for Short-Reach Optical Communications: A Review of Current Technologies and Future Trends," *Journal of Lightwave Technology*, vol. 36, pp. 377–400, Jan. 2018.
[34] E. Liang and J. Kahn, "Geometric Shaping for Distortion-Limited Intensity Modulation/Direct Detection Data Center Links," *IEEE Photonics Journal*, vol. 15, no. 6, pp. 1–17, 2023.
[35] G. P. Agrawal, *Fiber-Optic Communication Systems*. Wiley-Interscience, a John Wiley &Sons, 3rd edition ed., 2002.
[36] A. D. Gallant and J. C. Cartledge, "Characterization of the Dynamic Absorption of Electroabsorption Modulators With Application to OTDM Demultiplexing," *Journal of Lightwave Technology*, vol. 26, pp. 1835–1839, July 2008.

Authorized licensed use limited to: Technical University of Denmark DTU Library. Downloaded on May 19,2025 at 08:41:32 UTC from IEEE Xplore. Restrictions apply.
© 2025 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted,
but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.