Bayesian Optimization for Nonlinear System Identification and Pre-Distortion in Cognitive Transmitters

Matheus Sena, M. Sezer Erkılınç, Thomas Dippon, Behnam Shariati, Robert Emmerich, Johannes Karl Fischer, and Ronald Freund

***Abstract*—We present a digital signal processing (DSP) scheme that performs hyperparameter tuning (HT) via Bayesian optimization (BO) to autonomously optimize memory tap distribution of Volterra series and adapt parameters used in the synthetization of a digital pre-distortion (DPD) filter for optical transmitters. Besides providing a time-efficient technique, this work demonstrates that the self-adaptation of DPD hyperparameters to correct the component-induced nonlinear distortions as different driver amplifier (DA) gains, symbol rates and modulation formats are used, leads to an improvement in transmitter performance. The scheme has been validated in back-to-back (b2b) experiments using dual-polarization (DP) 64 and 256 quadrature amplitude modulation (QAM) formats, and symbol rates of 64 and 80 GBd. For DP-64QAM at 64 GBd, it is shown that the DPD scheme reduces the required optical signal-to-noise ratio (OSNR) at a bit error ratio of 10^{-2} by 0.9 dB and 0.6 dB with respect to linear DPD and a heuristic nonlinear DPD approach, respectively. Moreover, we show that the proposed approach also reduces filter complexity by 75% in conjunction with the use of memory polynomials (MP), while achieving a similar performance to Volterra pre-distortion filters.**

***Index Terms*—Bayes methods, Gaussian processes, nonlinear filters, optical transmitters, optimization.**

I. Introduction

With the increase of demand in bandwidth flexibility [1] in optical networks, there has been an emerging need for efficient resource management tools in order to address the diversity of capacity and reach demands required by users and services. Viable solutions rely on incorporating cognitive features into network operation, that, quoting [2], "… perceive current conditions, and then plan, decide, and act." In particular, significant progress has been recently achieved with the implementation of autonomous transponders [3]. These devices perform the self-adaptation of transmission parameters (e.g., modulation format, symbol rate, forward error correction (FEC) coding schemes) to automatically correct variable channel conditions originated from fiber and, especially, transceiver impairments (e.g., in-phase and quadrature (IQ) skew, IQ imbalance [4] and nonlinearities [5]). Alternatively, due to the advances of modern digital signal processing (DSP) tools, another promising solution to integrate cognition is via the self-adaptation of digital pre-distortion (DPD) schemes for optical transmitters [6], [7].

In the most general sense, DPD consists of building a transmitter nonlinear model that can be used to synthesize a DPD filter (e.g., by using an indirect learning architecture (ILA) [8]), in turn, employed to compensate for component-induced distortions. Conventionally, Volterra series is used as the nonlinear model because of its ability to capture memory effects [9], which in a DPD filter is embodied in a memory tap distribution. However, despite the effectiveness of Volterra-based DPD methods, as transmission becomes more dynamic and demanding (symbol rates $\geq$ 60 GBd coupled with advanced modulation formats), optimizing the total number of orders and memory taps of the Volterra filter without increasing the complexity of the model becomes a challenging task. Common estimation approaches use empirical tuning [7], requiring manual configuration of the memory tap distribution, or heuristic procedures inspired by grid-search [10], which are computationally expensive and, hence, inappropriate for cognitive applications. Optionally, one way to address this optimization problem is via modern Machine Learning (ML) algorithms.

ML has been claimed as a fundamental building block for the future of cognitive optical networks because its algorithms can learn from data, identify patterns and make decisions with minimal human intervention [11]. More importantly, ML-based algorithms have shown great compatibility to solve standard optical communication problems, while reducing complexity of traditional approaches [12]. In this regard, we envision strong similarities between the optimization of the orders and memory taps of a Volterra filter for DPD of optical transmitters and a design problem often coped within ML applications, so-called hyperparameter tuning (HT) [13]. The HT consists of an optimization process to search for ideal system parameters that

Manuscript received February 27, 2021; revised May 4, 2021; accepted May 17, 2021. Date of publication May 25, 2021; date of current version August 2, 2021. This work was funded by the EU Horizon 2020 Research and Innovation Program under MSCA-ETN WON, Grant 814276. *(Corresponding author: Matheus Sena.)*

Matheus Sena, M. Sezer Erkılınç, Behnam Shariati, Robert Emmerich, Johannes Karl Fischer, and Ronald Freund are with the Fraunhofer Institute for Telecommunications Heinrich-Hertz-Institute, Einsteinufer, 10587 Berlin, Germany (e-mail: matheus.sena@hhi.fraunhofer.de; sezer.erkilinc@hhi.fraunhofer.de; behnam.shariati@hhi.fraunhofer.de; robert.emmerich@hhi.fraunhofer.de; johannes.fischer@hhi.fraunhofer.de; ronald.freund@hhi.fraunhofer.de).

Thomas Dippon is with Keysight Technologies GmbH, Herrenberger Straße, 71034 Böblingen, Germany (e-mail: thomas_dippon@keysight.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JLT.2021.3083676.

Digital Object Identifier 10.1109/JLT.2021.3083676

maximize the accuracy (denoted by an objective function) of ML black-box models (e.g., artificial neural networks (ANN)). To solve the HT problem, Bayesian optimization (BO) is a Gaussian process (GP) based algorithm popularly chosen because of its sequential design strategy that facilitates the estimation of the global optimum in numerically expensive functions [14]. Moreover, BO offers reliable robustness and reduced convergence time, thus, supporting real-time automation in cognitive transmitters (CT).

In the framework of optical coherent systems, solving the HT problem via BO can be actually performed in two key stages of the DPD task, namely (1) nonlinear system identification (SI) and (2) synthetization of the DPD filter. In (1), hereafter referred to as Bayesian-based SI, the number of orders and memory taps of the Volterra filter that models the optical transmitter are optimized as means to improve the characterization accuracy. In (2), BO supports the synthetization of the DPD filter by tuning hyperparameters used in the ILA, in this paper referred to as Bayesian-based ILA.

This work is an extension of our previous contribution [15], in which we show that this approach enables autonomous identification and mitigation of transmitter impairments, helps reduce filter complexity and improves system level performance. This study extends the results reported in [15] by:

- deepening the level of discussion on the technical details of BO,
- presenting the computational gain that justifies the use of the proposed approach,
- experimentally assessing Bayesian-based SI and Bayesian-based ILA under different setup configurations and DPD filter design scenarios.

The remainder of this text is structured as follows. Section II briefly reviews other related works and clarifies this manuscript's position. Section III provides an introduction to the BO algorithm. Section IV explains how we incorporate the concept of HT to formulate the Bayesian-based SI and ILA. In Section V, we show the computational performance of the BO algorithm and demonstrate the benefits of using Bayesian-based SI and Bayesian-based ILA in a back-to-back setup by varying three configuration setups, such as DA gain, symbol rate (64 and 80 GBd) and modulation format (dual-polarization (DP) 64 and 256 quadrature amplitude modulation (QAM)). Still in Section V, the approach is benchmarked and validated with the use of memory polynomial (MP) DPD filters [16], in which we show that significant filter complexity reduction can be achieved. At last, the main conclusions and considerations are summarized in Section VI.

Important notations and concepts. Throughout this manuscript, vectors and matrices are written with bold lowercase and uppercase letters, respectively. Each individual element of a vector or matrix is denoted by $[\cdot]_{x,y}$, where the subscripts x and y correspond to the row and column number, respectively. The transpose of a vector is represented by $(\cdot)^T$, while the inverse of a matrix is written as $(\cdot)^{-1}$. In this work we also use some theoretical concepts of stochastic processes, such as GP, which is defined as a collection of random variables, where any finite set of these variables has a joint Gaussian distribution [17]. Furthermore, we recurrently express the conditional probability density function of a random variable Y given the occurrence of X as $p(Y|X)$.

II. Related Work

ML-based techniques for transmitter [18], [19], receiver [20] or link [21] impairment compensation in communication systems date back to the 1990s. Initially designed for digital radio communication, these schemes mostly consisted of training ANNs to learn the generalization of component/medium-induced distortions, which permitted the mitigation of impairments that cannot be analytically modeled. In the field of optical communications, the use of ML approaches has gained massive attention over the past decade, predominantly targeting tasks, such as fiber and transceiver nonlinearity mitigation [12], [22]–[24], but also extensively explored in optical performance monitoring techniques [25], [26] and network resource allocation strategies [27].

In the context of optical transmitters, the use of ML methods for impairment mitigation through DPD has been considered of paramount importance, recently showing the potential of breaking records in increasing net rate for single-channel transmission [28]. These approaches have also been regarded as an alternative to conventional Volterra series [7], which assumes a transmitter model, while ML techniques tend to be more model-free. In [29], for instance, the authors propose and demonstrate with simulations a two-step learning approach for neural network based DPD, which is applied to a Mach-Zehnder modulator (MZM) based coherent transmitter and presents advantages in performance with respect to Volterra filters, especially in cases of strong nonlinearity and noise conditions. Yet, one of the general pitfalls of applying neural networks in such tasks is their dependence on proper training, which includes the correct selection of datasets as well as efficient tuning of model parameters, meaning that their use is not straightforward and several issues can be encountered, even when the algorithm is correctly implemented [30]. This directly impacts the fast adaptability of the DPD schemes, which is a required feature in dynamic and heterogeneous network scenarios, where transmission parameters (e.g., modulation format, symbol rate, forward error correction (FEC) coding schemes) and device technologies are diverse, thus demanding real-time reconfiguration of the compensation scheme. With the purpose of providing more adaptability to ML-based DPD, other works have targeted approaches that can produce good generalization performance, while learning in a time-efficient manner [31], [32]. In [31], for instance, the authors introduce a low-complexity memoryless scheme based on Extreme Learning Machine, enabling rapid compensation of MZM transfer functions, which permits its use regardless of the modulator technology.

A common characteristic in the above-cited works is that the application of ML algorithms mostly concerns the better estimation of the filter coefficients and do not deepen the analysis on the optimization of the filter architecture. Discussions related to this topic have already been briefly initiated for DPD in [33] and described in more detail for post-equalizers in [24].

In both works, the authors demonstrate the potential of using ML optimization tools, such as BO, to automate the search for optimal designs of neural networks that can compensate for transceiver and fiber noise. This triggers the potential of using BO to enhance the design of Volterra-based DPD filters, thus tackling scalability pitfalls and alternatively replacing conventional pruning/truncation techniques such as regularization [34] or grid-based heuristic methods [10]. Therefore, this work proposes a scheme that uses traditional Volterra series for DPD assisted by a modern ML-based optimization tool, i.e., BO. Unlike other previous works, the proposed approach permits to automatically estimate the design of Volterra-based DPD filters. By doing so, this work addresses scalability issues through the optimizations of the model complexity, while proving to be a time-efficient and hence a suitable method for CT.

III. Hyperparameter Tuning via Bayesian Optimization

For decades, black-box models have attracted much attention in both academia and industry due to their efficiency on characterizing complex nonlinear systems. Nevertheless, an important challenge limiting the performance of these models is their intrinsic dependence on the selection of hyperparameters, i.e., any model parameter that can be set beforehand to control learning algorithms [35]. Hyperparameters affect the speed and accuracy of the learning process of black-box models and, in contrast with conventional model parameters, hyperparameters cannot be easily estimated from the dataset. Moreover, the lack of analytical formulas to calculate hyperparameters and the restricted options of methods (e.g., exhaustive and random search) have driven interest in more sophisticated optimization techniques over the past years. One of these techniques is the BO, which is reviewed in the following sub-section.

A. Bayesian Optimization and the Problem Statement

Given a black-box model that characterizes a system under test (SUT), in which a given arbitrary input s yields a response r, then the model accuracy can be evaluated through an objective function f. A hyperparameter entry, represented by a scalar (or vector) input θ, determines this evaluation, such that $f = f(\theta, s, r)$, which for simplicity can be written as $f = f(\theta)$, $f: \Theta \rightarrow \mathbb{R}$. In order to estimate the optimal model accuracy, f must be subject to an optimization process with respect to θ. However, in most cases, this optimization of f is bounded by two important restrictions, they are:

1) *Computational complexity* – The number of evaluations performed on f is limited, typically in the range of a few hundreds. This condition frequently arises because each evaluation takes a substantial amount of time.
2) *Non-differentiability* – First- and second-order derivatives of f with respect to θ, i.e., $f'(\theta)$ and $f''(\theta)$, are not obtainable, thus, preventing the application of methods like gradient descent, Newton's method, or quasi-Newton methods.

In summary, the maximization of f is defined as a HT problem and can be mathematically written as:

$$\underset{\theta \varepsilon \Theta}{\text{maximize}}\, f(\theta) \tag{1}$$

where Θ is known as hypothesis space (HS) and represents the domain of hyperparameters that can be numerically evaluated in f. In order to solve this mathematical problem, BO is a promising solution because it suppresses the aforementioned restrictions, i.e., it needs relatively few evaluations on f and it is a derivative-free method.

BO logically depends on two core principles. First, it builds a basic *surrogate function* f^* to "fit" the objective f and estimate its response to unknown entries θ. Second, it bypasses the impossibility of using gradient descent methods on f by introducing an *acquisition function*, i.e., a statistical operator that orients the optimum search.

B. Surrogate Function

One of the fundamental ideas of the BO is the capability to iteratively create a surrogate function $f^* = p(f|\mathcal{D})$ that estimates the value of the objective function f for arbitrary θ, i.e., $f(\theta)$, conditioned on a limited sub-set of n-observed data points ($\mathcal{D} = \{f(\theta_1), f(\theta_2), \ldots, f(\theta_n)\}$). To build f^*, the BO algorithm models $p(f|\mathcal{D})$ as a GP, which permits to represent the posterior distribution $p(f|\mathcal{D})$ by the normal distribution $\mathcal{N}(\mu, \sigma^2)$, where $\mu: \Theta \rightarrow \mathbb{R}$ and $\sigma^2: \Theta \rightarrow \mathbb{R}$ correspond to a mean and a variance function, respectively. The main benefit of using GP is the possibility to apply algebraic properties that when incorporated into Bayes's rule enable us to analytically write $\mu(\theta)$ and $\sigma^2(\theta)$ as [14]:

$$\mu(\theta) = \mathbf{k}^T(\theta)\,\mathbf{K}^{-1}\mathbf{z}, \tag{2}$$

$$\sigma^2(\theta) = k(\theta, \theta) - \mathbf{k}^T(\theta)\,\mathbf{K}^{-1}\mathbf{k}(\theta). \tag{3}$$

Eq. 2 and 3 are fully determined by the kernel covariance function $k: \Theta \times \Theta \rightarrow \mathbb{R}$, the n-by-1 vector $\mathbf{z}$ and the n-by-n Gram matrix $\mathbf{K}$. The kernel covariance function $k(\theta, \theta')$ is built by applying a covariance function between two arbitrary entries $\theta, \theta' \in \Theta$, i.e., $k(\theta, \theta') = Cov(\theta, \theta')$, consequently yielding the 1-by-n vector $\mathbf{k}^T(\theta)$, where $[\mathbf{k}(\theta)]_{u,1} = Cov(\theta, \theta_u)$ for $u \in \{1, \ldots, n\}$, and the scalar $k(\theta, \theta) = Cov(\theta, \theta)$. Finally, the vector $\mathbf{z}$ and the Gram matrix $\mathbf{K}$ are respectively defined as $[\mathbf{z}]_{u,1} = f(\theta_u)$ and $[\mathbf{K}]_{u,v} = k(\theta_u, \theta_v)$, where $u, v \in \{1, \ldots, n\}$.

C. Acquisition Function

The acquisition function a can be evaluated for any arbitrary input θ and quantifies how promising the next sampling decision θ_{n+1} is to indicate the location of the global optimum. By maximizing the acquisition function, i.e., $\theta_{n+1} = \underset{\theta \varepsilon \Theta}{\text{maximize}}\, a(\theta)$, to select the next numerical evaluation $f(\theta_{n+1})$, we merely substitute our initial optimization problem (Eq. 1) with another optimization, but now with a cheaper function. A common choice for the acquisition function is the expected improvement

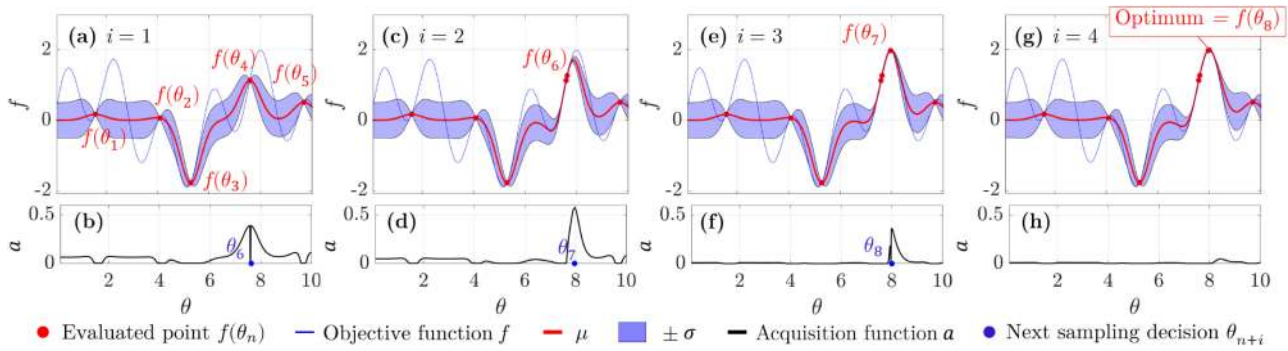

Fig. 1. (a) Initialization of the surrogate model with $n = 5$ observations $\mathcal{D} = \{f(\theta_1), \ldots, f(\theta_5)\}$. (a, c, e, g) Updated surrogate function for $i = 1$, 2, 3 and 4, respectively. (b, d, f, h) Acquisition function for $i = 1$, 2, 3, and 4, respectively. In this example, the kernel length γ is set to 0.5.

(EI), computed as [14]:

$$a\ (\theta) = \left[\mu\,(\theta) - f^{+}\right]\ \Phi\,(\mathrm{Z}) + \sigma\,(\theta)\,\phi\,(\mathrm{Z})\,, \tag{4}$$

where $f^{+} = \max(\mathcal{D})$ and $Z = \frac{\mu(\theta) - f^{+}}{\sigma(\theta)}$ if $\sigma(\theta) > 0$ or $Z = 0$ if $\sigma\ (\theta) = 0$. The functions Φ and ϕ correspond to the cumulative and probability density functions of the standard normal distribution $\mathcal{N}(0, 1)$, respectively. Since $a(\theta)$ can be analytically expressed as a function of $\mu(\theta)$, $\sigma(\theta)$ and f^{+}, which are directly obtained from the surrogate function f^{*}, the sampling point θ_{n+1} is easily found by numerically evaluating $a(\theta)$ for all $\theta \in \Theta$. The EI is used in all test cases along this work.

D. Choosing a Kernel Covariance Function

To build the *surrogate* function, the GP needs to consider some assumptions about the shape of the function f. This shape assumption is incorporated in the choice of the kernel covariance function $k\,(\theta, \theta') = Cov(\theta, \theta')$. In this work, we use a universal kernel covariance function, the so-called Gaussian radial basis function, which can be analytically expressed as:

$$Cov\ (\theta, \theta') = \exp\left(-\frac{\|\theta - \theta'\|^2}{2\gamma^2}\right)\,, \tag{5}$$

where $\|\,\|^2$ represents the l-2 norm and γ the kernel length. The kernel length γ is per se a hyperparameter of the GP and determines the smoothness of k. A rapid and reliable estimation approach to determine γ is by using the strategy presented in [17]. There, the authors propose that γ should be selected such to maximize the log marginal likelihood (LML) function (Eq. 5.8 in [17]), which can be expressed as: $\mathrm{LML} = -\frac{1}{2}(\mathbf{z}^T\mathbf{K}^{-1}\,\mathbf{z}) - \frac{1}{2}\log|\mathbf{K}| - \frac{n}{2}\log 2\pi$. Since $\mathbf{K}$ is a function of γ, i.e., $\mathbf{K} = \mathbf{K}(\gamma)$, then LML is also a function of γ, that is, $\mathrm{LML} = \mathrm{LML}(\gamma)$, which permits us to perform this maximization. To find a good estimation for the γ that maximizes the LML in the experiments carried out in this work, we executed a coarse search by numerically evaluating the LML in the interval $\{\gamma \mid 0 < \gamma \leq 20\}$, with steps $\Delta\gamma = 1$. The complexity of computing the LML is dominated by the need to invert the n-by-n matrix $\mathbf{K}$, which scales the time by $\mathcal{O}(n^3)$. However, on a conventional computer with 16 GB of memory RAM for $n = 50$ (maximum number of evaluations assumed in this work) this operation is in the range of 200 ms, still proving to be computationally fast to justify its use.

Algorithm 1: Pseudo-code for BO.

Input: Θ
Output: θ_{opt}
1: Initialize $\mathcal{D} = \{f(\theta_1), f(\theta_2), \ldots, f(\theta_n)\}$
2: while $i \leq Max_{it}$ do
3: Compute f^{*}
4: Maximize a
5: Sample $f(\theta_{n+i})$
6: Increment $\mathcal{D} = \{f(\theta_1), \ldots, f(\theta_{n+i})\}$
7: **end while**
8: $\theta_{opt} = \arg\max\,(\mathcal{D})$

E. The Bayesian Optimization Algorithm

To summarize this brief review, the BO algorithm can be defined by the pseudo-code in Algorithm 1. In line 1, the training set $\mathcal{D}$ is initialized by sampling f at n points in Θ. It should be noted that this sampling can be performed either randomly, when no previous information is known about f, or deterministically, when there is some indication about the optimum of f. Then, in line 2, the BO is programmed to run until a maximum number of iterations Max_{it} is reached. For each i-th iteration loop, the *surrogate* function f^{*} is computed (line 3), i.e., μ and σ^2 are calculated, and used to maximize an *acquisition* function a (line 4), which, provides a new sampling decision θ_{n+i}. Finally, the sampling decision is evaluated $f(\theta_{n+i})$ (line 5) and incorporated to $\mathcal{D}$ (line 6) before a new cycle starts. After this iterative process ends (line 8), the hyperparameter θ that yields the maximum $f(\theta)$ in $\mathcal{D}$ is selected as the optimal solution θ_{opt}.

Algorithm 1 is also illustrated with an example in Fig. 1. Let's suppose the maximization problem of the function $f(\theta) = \sin(\theta) + \sin(10\theta/3)$, $f : [0, 10] \rightarrow \mathbb{R}$ (blue curve in Fig. 1a). As it can be inspected from Fig. 1a, an initial set of randomly evaluated points $\mathcal{D} = \{f(\theta_1), \ldots, f(\theta_5)\}$ (red dot markers) permits us to construct the surrogate function f^{*}, depicted by μ (red solid line) and a confidence interval $\mu \pm \sigma$ (blue-shaded area).

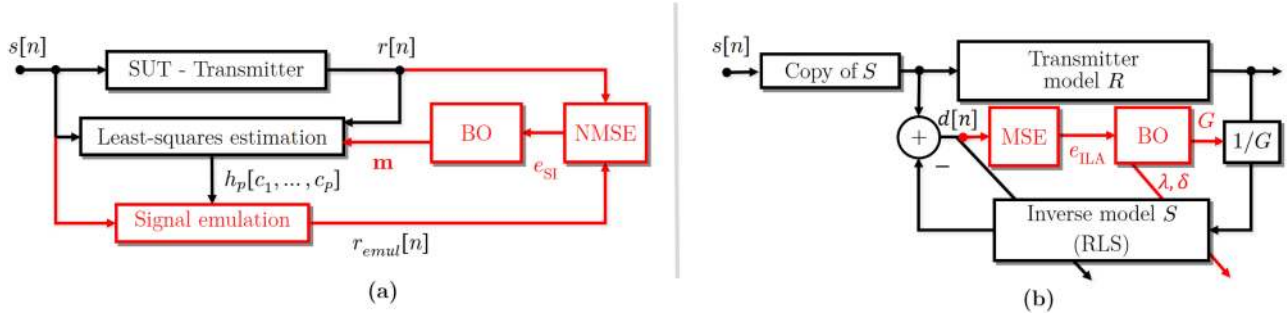

Fig. 2. (a) Bayesian-based SI for autonomous tuning of the SUT filter design ($\mathbf{m}$) and (b) Bayesian-based ILA for optimization of loop parameters (λ, σ, G). The red-colored paths and boxes represent the incorporation of the HT via BO into traditional SI and ILA approaches (black paths and boxes).

To infer the next evaluation, the *acquisition* function (Fig. 1b) is maximized and θ_6 is acquired (blue dot marker). In the sequence, $f(\theta_6)$ is numerically evaluated (Fig. 1) and incorporated to $\mathcal{D}$. Then, this process is repeated, i.e., the *surrogate* function is updated (Fig. 1c, e, g) and followed by the maximization of *acquisition* function (Fig. 1d, f, h) to provide the new sampling decisions. Notably, very few iterations are needed to identify the global maximum of f, as shown in Fig. 1g. This evidences the potentiality of BO to reduce convergence time, which is a crucial requirement in the design of CT.

After this brief review on HT via BO, we introduce how this technique can be explored in DPD of optical transmitters.

IV. Bayesian Optimization in Digital Pre-Distortion of Optical Transmitters

Volterra-based DPD of optical transmitters can be defined in two major stages [7]:

1) *System Identification (SI)* – A Volterra series is used to build a model R of the SUT based on measurements of the input waveform s and output waveform r.
2) *Signal Pre-Distortion* – An inverse model S of the SUT is synthesized and operates on s to generate a pre-distorted signal $\hat{s}$. When $\hat{s}$ is applied to the SUT, the output waveform is s', which in ideal conditions, $s' = s$.

These two DPD stages heavily rely on hyperparameters, as shown in the next sub-sections, and their improper tuning may lead to suboptimal transmission performance. For that reason, we incorporate BO into DPD.

A. Bayesian-Based SI

Conventionally, an optical transmitter consists of DACs followed by a quad set of DAs and a DP IQ-modulator. The transmitter output is a continuous-time waveform that when represented in its discrete form, i.e., $r[n]$, can be modeled with respect to the input $s[n]$ via a truncated, time-invariant Volterra series, such that [7]:

$$r\ [n] = h_0\ + \sum_{p\ =\ 1}^{P} \sum_{c_1 =\ 0}^{m_p - 1} \cdots \sum_{c_p =\ c_{p-1}}^{m_p - 1} h_p\left[c_1, \ldots, c_p\right] \times \prod_{i\ =\ 1}^{p} s\left[n - c_i - \tau_p\right]. \tag{6}$$

Since in any physically realizable system the output can only depend on present and previous values of the input, what is known as the causality condition, τ_p in Eq. 6 is an arbitrary delay used for non-causal filtering realizations, i.e., it aligns the higher order kernels centric to the first order kernel. The Volterra series is most importantly determined by the kernel coefficients $h_p[c_1, \ldots, c_p]$ with memory lengths m_p, where $p\ \varepsilon\ \{1, \ldots, P\}$ represents the order of the kernel. When performing SI, h_p can be estimated via adaptive algorithms (e.g., least-squares estimation [36]) as depicted in Fig. 2a, where blocks of samples from $s[n]$ and $r[n]$ are used to obtain h_p. However, in this scheme, the accuracy of the obtained Volterra model strongly depends on the hyperparameter $\mathbf{m}\ =\ [m_1, \ldots, m_p, \ldots, m_P]$. To facilitate the assimilation of h_p and $\mathbf{m}$, we show in Fig. 3(a-c) a visual representation of the normalized first (h_1), second (h_2) and third (h_3) order Volterra kernel coefficients for a synthetic 5$^{\text{th}}$-order filter, where $\mathbf{m} = [39, 5, 7, 3, 1]$.

Briefly, in the SI process the least-squares algorithm is used to estimate a Volterra filter that optimally fits the SUT input signal ($s[n]$) to the SUT measured output ($r[n]$). After that, an emulated SUT response ($r_{emul}[n]$) can be obtained by applying the resulting Volterra filter to $s[n]$. The similarity between $r_{emul}[n]$ and $r[n]$ heavily relies on $\mathbf{m}$ (the memory vector), which is considered a hyperparameter because it needs to be set beforehand to model the Volterra filter in which the p-th order kernel coefficients h_p are estimated. Therefore, to incorporate BO into SI, we restructure the conventional SI block diagram Fig. 2a to search for the $\mathbf{m}_{opt}$ that minimizes the identification error e_{SI}. In this work, we quantify e_{SI} by the normalized mean squared error (NMSE) of the emulated signal $r_{emul}[n]$ with respect to the measured output signal $r[n]$ [37]:

$$e_{\text{SI}}\left(\mathbf{m}\right)\ \triangleq\ \frac{Var\left(r_{emul} - r\right)}{Var\left(r\right)}, \tag{7}$$

where $Var(\cdot)$ denotes the variance. Since $1 - e_{\text{SI}}(\mathbf{m})$ is a proper figure of merit (FOM) of the SUT model accuracy, we write the HT problem, such that:

$$\underset{\mathbf{m}\ \varepsilon\ \mathcal{M}}{\text{maximize}}\ 1 - e_{\text{SI}}\left(\mathbf{m}\right) \tag{8}$$

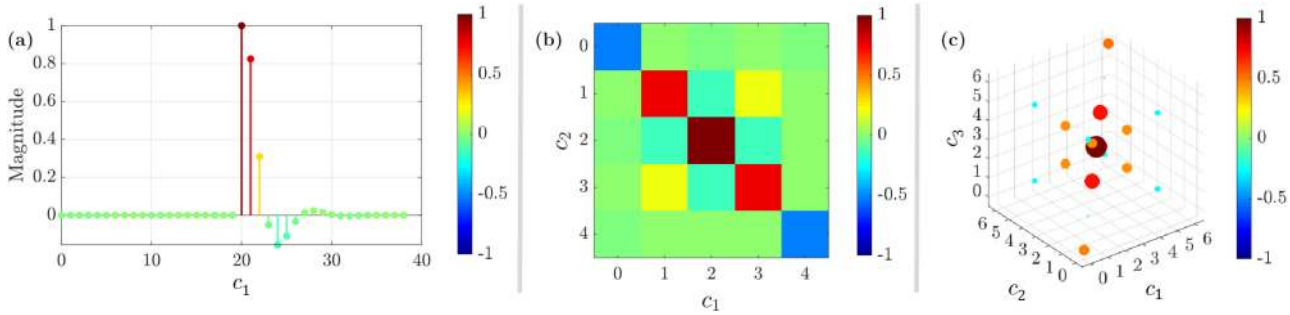

Fig. 3. Normalized (a) first (h_1), (b) second (h_2), and (c) third (h_3) order Volterra kernel coefficients of a 5th-order synthetic filter for $\mathbf{m} = [39, 5, 7, 3, 1]$.

subject to

$$M_c \geq \sum_{p\,=\,1}^{P} \frac{(m_p + p - 1)!}{(m_p - 1)!p!}, \tag{8.1}$$

$$m_1 > \; m_p \; \forall \, p \geq 2, \tag{8.2}$$

$$\mathcal{D}_{\mathrm{SI}} = 1 - e_{\mathrm{SI}}\left([1, \ldots, \, 0, \ldots, \, 0]\right)\}. \tag{8.3}$$

$\mathcal{M}$ is the HS of memory vectors and can be represented by a P-dimensional hypercube $\mathbb{R}^P$, such that each element $\mathbf{m}$ in $\mathcal{M}$ satisfies a maximum number of computed filter coefficients M_c (filter complexity), denoted by Eq. 8.1 [38]. To illustrate, for $\mathbf{m} = [10, \, 10, \, 10]$, a direct inspection of Eq. 8.1 shows that $\sum_p \frac{(m_p+p-1)!}{(m_p-1)!p!} = 10 + 55 + 220 = 285$, where one can clearly see that for a fixed memory length, higher orders generate more kernel coefficients. It is also important to highlight that despite the total number of coefficients of a Volterra series is $\sum_p m_p{}^p$, the number of coefficients that actually have to be estimated is smaller, since the products in Eq. 6 remain the same when two different indices are permuted (e.g., for $p = 2$ $h_2[c_1, \, c_2] = h_2[c_2, \, c_1]$). Due to this symmetry, the required number of computed coefficients can be expressed as in Eq. 8.1. For the analysis derived in this work, we considered $P = 5$, that is, $\mathcal{M}$ covers memory effects up to the 5$^{\text{th}}$ order, or more simply, $\mathbf{m} = [m_1, \, m_2, \, m_3, \, m_4, \, m_5]$. Furthermore, we defined that the 1$^{\text{st}}$-order memory taps (m_1) must be greater than the nonlinear ones ($m_p, \forall \; p \geq 2$) in order to generate physically realistic filter designs and create a more compact $\mathcal{M}$ (Eq. 8.2). At last, since we assume that no *a priori* information is known about the optical transmitter, the training set $\mathcal{D}_{\mathrm{SI}}$ is initialized with a simple single-tap 1$^{\text{st}}$-order Volterra filter (Eq. 8.3). This assumption also helps comparing the proposed technique with an alternative benchmark later introduced in this text. We name this optimization process Bayesian-based SI and it is depicted by the full block diagram of Fig. 2a. The benefit of using BO to find $\mathbf{m}_{opt}$ is that at the end of each n-th iteration, the next sampling decision $\mathbf{m}_{n+1}$ in $\mathcal{M}$ is obtained from the maximization of a cheap acquisition function, as introduced in sub-section III.C, permitting to reach the global optimum in a gradient-free fashion.

As it can be noticed, the Bayesian-based SI tries to find the architecture that best describes the forward path of the black box system, i.e., from the SUT input to the SUT output, where the $\mathbf{m}_{opt}$ that yields an optimal h_p is bounded to a maximum filter complexity constraint M_c. This complexity constraint is an important design parameter that gives flexibility to tailor the size of the transmitter DPD filter according to the project specifications (e.g., available on-chip memory). In Section V, we suggest a methodology to identify an optimal value for M_c.

B. Bayesian-Based ILA

The ILA is often utilized for synthesizing the inverse Volterra model S. A block diagram of the employed architecture is shown in Fig. 2b (black boxes and paths). In this scheme, to pre-compensate a SUT described by its model R, the algorithm iteratively generates a new prototype pre-distortion filter S by minimizing the error signal $d[n]$. The major benefit of the ILA is the fact that the derived pre-distortion filter S is applicable to arbitrary input signals other than the training signal. Moreover, the ILA does not rely on a pre-defined architecture of the distorting system R.

In the process of minimizing the ILA error vector $d[n]$ (Fig. 2b), a recursive least-squares (RLS) algorithm is used [39]. The performance of the RLS algorithm relies on important hyperparameters, i.e., the forgetting factor λ and the gain factor δ, which control the accuracy and speed of the algorithm, respectively. Finally, another critical issue in ILA is the selection of the normalization gain G applied in the feedback path before S (Fig. 2b). Therefore, given the ability of HT to release the recursive process from the necessity of manually tuning these hyperparameters, BO is also extended to ILA to optimize the computation of the inverse filter S. In this HT problem, the goal is to find the vector $\mathbf{w} = [\lambda, \, \delta, \, G]$ that minimizes $e_{\mathrm{ILA}}(\mathbf{w})$, which is computed by the mean squared-error (MSE):

$$e_{\mathrm{ILA}}\left(\mathbf{w}\right) \triangleq \frac{1}{L} \sum_{n\,=\,1}^{L} \left(d\left[n\right]\right)^2, \tag{9}$$

where L is the total number of samples in the error vector $d[n]$. In this work, we used $L = 8192$. Given that $1 - e_{\mathrm{ILA}}(\mathbf{w})$ is a FOM for the accuracy of the filter synthetization process, we

write this optimization problem as:

$$\underset{\mathbf{w}\ \varepsilon\ \mathcal{W}}{\text{maximize}}\ 1 - e_{\text{ILA}}\left(\mathbf{w}\right) \tag{10}$$

subject to

$$\mathcal{D}_{ILA} = 1 - e_{\text{ILA}}\left([\lambda_1,\ \delta_1,\ \ G_1]\right)\}. \tag{10.1}$$

To define the boundaries of $\mathcal{W}$, we use a common RLS strategy that suggests that good values for λ are close to, but not equal to 1, whereas δ is commonly 100-1000 times the variance of the input sequence and $G \leq 1$. Therefore, in this work we set $\lambda,\ \delta$ and G such that $\{\lambda \mid 0.99 \leq\ \lambda\ \leq 0.9999999\}$, $\{\delta \mid 100 \leq\ \delta\ \leq 1500\}$ and $\{G \mid 0.5 \leq\ \delta\ \leq 1\}$. The training set, i.e., $\mathcal{D}_{ILA}$, was initially built by evaluating $1 - e_{\text{ILA}}\left([\lambda_1,\ \delta_1,\ \ G_1]\right)$, where $\mathbf{w} = [\lambda_1,\ \delta_1,\ \ G_1]$ was randomly sampled in $\mathcal{W}$. This optimization process is named Bayesian-based ILA and summarized by the block diagram in Fig. 2b. Analogously to the Bayesian-based SI, the next sampling decision $\mathbf{w}_{n+1}$ in $\mathcal{W}$ is obtained from the maximization of the acquisition function, as introduced in sub-section III.C, permitting to reach the global optimum $\mathbf{w}_{opt}$ without computing gradients.

It is noteworthy that, in this manuscript, we assume the filter architecture of the model S as equal to the filter architecture of the model R (with an exception to the results shown in section V.F.3), since the latter captures the necessary memory effects that optimally model the transmitter and, therefore, it is sufficient to compensate for the SUT impairments.

V. Results and Discussions

In this section we discuss the benefits of using the Bayesian-based SI and Bayesian-based ILA in the context of optical coherent systems. First, in sub-section A, we demonstrate the computational gain associated with the use of Bayesian-based SI in comparison to a heuristic memory tap optimization approach. Then, in sub-section B, we introduce the testbed and procedures used in our experimental validations and propose a methodology for the autonomous operation of the DPD scheme. In sub-sections C-E, we test our approach for different setup configurations. Finally, in sub-section F, we validate the optical performance of the technique comparing it to other DPD filter design scenarios.

A. Computational Gain

Recently, a SI approach was introduced in [10]. In this scheme, a heuristic was proposed to optimize the number of orders and memory taps of the Volterra series and follows the described method. First, the heuristic initializes the SI with a single-tap 1st-order Volterra filter, i.e., the least-squares estimation to obtain h_p is carried out for a memory vector $\mathbf{m}\ =\ [1]$. Then, m_1 is unitarily incremented until e_{SI} reaches an error floor. After that, m_1 is fixed and this procedure repeated for higher orders (e.g., $m_2,\ m_3, \ldots$). The algorithm stops when an optimal e_{SI} is found after adding new orders. Despite providing an accurate estimation, this approximation gives rise to a difficult implementation constraint, i.e., the high number of evaluations of e_{SI} to find $\mathbf{m}_{opt}$, which arises from the grid-search-like nature of the assumed strategy. Unlike the Bayesian-based SI, the heuristic introduced in [10] performs an uninformed search by operating in a brute-force way and does not efficiently utilize the information from the intermediate states tested during the optimization process.

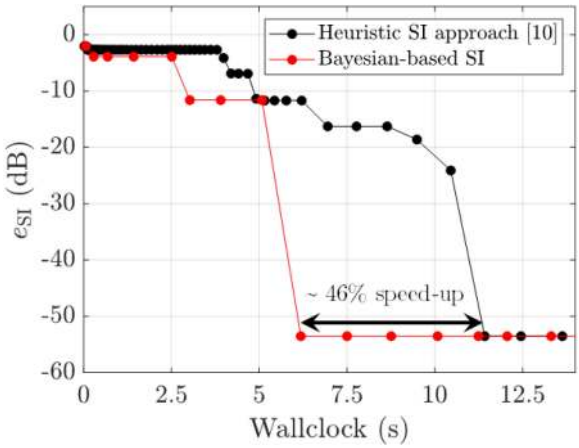

Fig. 4. Bayesian-based SI is benchmarked with the heuristic memory tap optimization approach introduced in [10]. A 46% speed-up is achieved by using the derived approach when both strategies are tested on a synthesized Volterra filter represented by the memory vector $\mathbf{m} = [39, 5, 7, 3, 1]$.

To benchmark Bayesian-based SI with [10], we synthesized a 5th-order Volterra model that emulates the response of a SUT and is depicted by Fig. 3, for which $\mathbf{m}$ is known and equal to $\mathbf{m} = [39, 5, 7, 3, 1]$. Then, a random eight-level pulse-amplitude modulation (PAM-8) training sequence ($s[n]$) with 10^5 symbols at 1 Sa/symbol is fed to the emulated SUT model to obtain the corresponding output signal ($r[n]$). This pair of waveforms, i.e., $s[n]$ and $r[n]$, is hence provided to the proposed Bayesian-based SI and to the heuristic approach. Then, both techniques are used to blindly learn the optimal $\mathbf{m}$. The SUT model was emulated with *VPItoolkit*$^{\text{TM}}$ *DSP Library*. To create the HS for the BO, i.e., $\mathcal{M}$, we set a maximum filter complexity of $M_c = 155$, which ensures that $\mathbf{m}_{opt}\ \varepsilon\ \mathcal{M}$. The Wallclock time (elapsed processing time) after each iteration loop for both techniques was then used as FOM. According to the results shown in Fig. 4, Bayesian-based SI is able to reach the minimum identification error e_{SI}, 46% faster compared to the benchmarked approach [10]. This indicates that using BO to identify the optimal memory tap distribution of a Volterra filter brings the advantage of reducing the convergence time, what makes it more computationally suitable for CT. It is also important to point out that both approaches can learn the exact m = [39, 5, 7, 3, 1], and that the non-null error floor comes from a negligible residual error of the least-squares estimation, which is used to obtain the kernel coefficients in both methods.

B. Experimental Testbed and the DPD Hyperparameter Validation

The experimental validation is composed of two parts: (1) SI and (2) performance evaluation of the DPD. Both parts were realized using the same experimental testbed, as shown in Fig. 5a and detailed in [40]. The methodology utilized in this

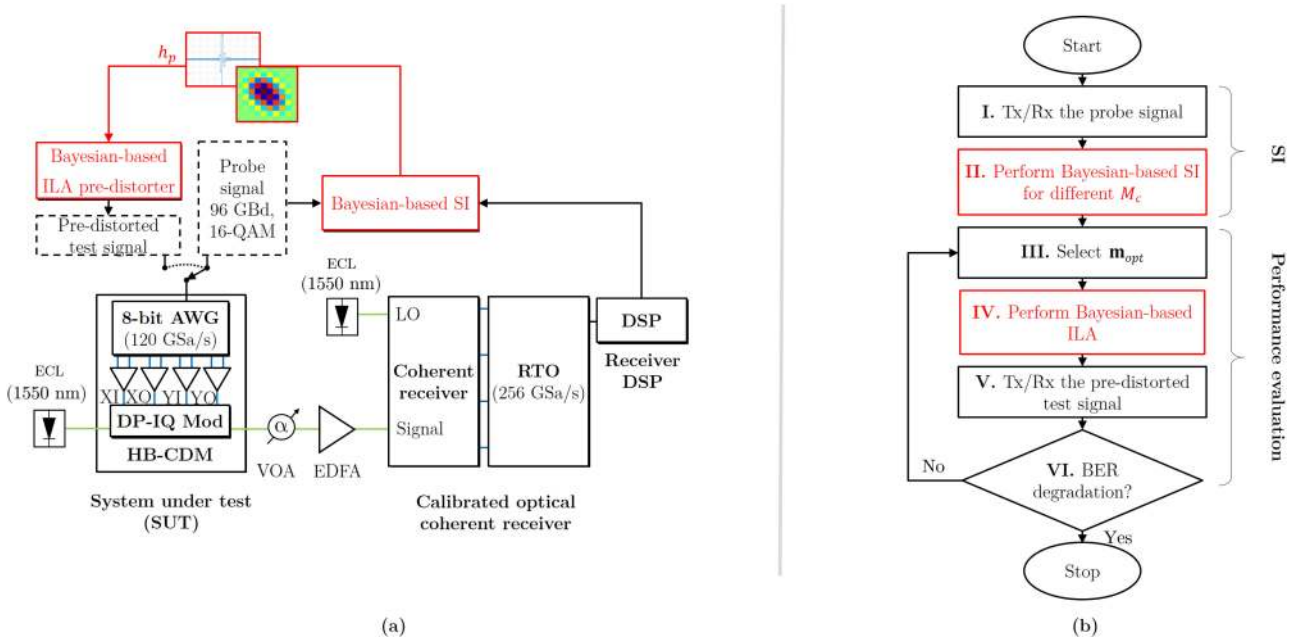

Fig. 5. (a) Experimental setup. (b) Block diagram for validation of the DPD hyperparameters. g_{DA}: driver amplifier gain.

validation is also logically described in the block diagram in Fig. 5b, which depicts how the Bayesian-based SI and ILA can be jointly handled for tuning of the DPD hyperparameters.

In the first part (i.e., SI), a 96 GBd DP-16QAM "probe" signal was generated using a 2^{15} random bit sequence followed by a root-raised cosine (RRC) pulse shaping filter with roll-off factor 0. The 96-GBd symbol rate ensures that the identification of the SUT Volterra model covers sufficient frequency components to guarantee that the test signals (at 64 and 80 GBd) will not be cut off in the frequency domain when applied to the inverse model S (Fig. 2). Additionally, the probe signal needs to excite multiple DAC discretization levels, which in theory can be performed with any modulation format with cardinality higher than two (e.g., 16QAM, 64QAM, 256QAM). However, it is also important to remind that during the SI the signal is not yet pre-distorted, which means that it contains transmitter IQ time-skew and amplitude imbalances. In these conditions, especially when a high symbol rate must is applied (96 GBd), a 16QAM signal is less susceptible to impairments when compared to higher cardinalities [41], incurring in a more resilient identification, what explains the use of this modulation format for the probe signal. The four sample sequences for the quadrature components (XI, XQ, YI and YQ) were then uploaded to a 4-channel 120 GSa/s Keysight arbitrary waveform generator (AWG) with 3-dB bandwidth of 45 GHz (able to generate signals with frequency components above 45 GHz). The AWG was used to drive a 40-GHz optical multi-format transmitter (OMFT) from ID Photonics based on a high-bandwidth coherent driver module (HB-CDM). Together, the OMFT and the AWG comprise the SUT as indicated in Fig. 5a. An external cavity laser (ECL) at fixed wavelength (1550 nm) was used for the DP-IQ modulator and for the local oscillator (LO). Then, the optical signal was transmitted, received and digitized using an optical coherent receiver (70 GHz) followed by a 256 GS/s real-time oscilloscope (RTO) with 110 GHz analog bandwidth (Fig. 5b-I). We removed the distortions originated from IQ-skew and the frequency response induced by the RTO by performing an additional offline calibration at 1550 nm. Then, the derived correction was applied to all the following measurements in the receiver DSP. At the receiver DSP, Stokes space based polarization demultiplexing [42], clock recovery, resampling, frequency offset correction [43] and carrier phase recovery [44] were performed. The received and the transmitted samples of the probe signal quadrature components were finally provided to the Bayesian-based SI. At this stage, the Bayesian-based SI was processed for different filter complexities M_c (Fig. 5b-II), at $Max_{it} = 50$ iterations, hence, generating different optimal designs $\mathbf{m}_{opt}$ and kernel coefficients h_p that were stored to be validated in the second part, i.e., performance evaluation of the DPD. As mentioned early in this text, M_c is an intrinsic hyperparameter of the Bayesian-based SI problem and depends on the project specification (e.g., available on-chip memory). However, here the reason to process Bayesian-based SI for multiple complexities is to demonstrate that there is a minimum M_c, beyond which the filter design enters the overfitting regime, i.e., when increasing model complexity yields degrading performance. This occurs when the BO interprets the noise in the received quadrature components as nonlinear distortions and tries to model them by adding unnecessary memory taps to the extent that it negatively impacts the transmission performance. The results for the Bayesian-based SI were obtained for a kernel length of $\gamma = 10$.

In the second stage (performance evaluation of the DPD), as illustrated in Fig. 5b III-VI, we used an iterative process to select the $\mathbf{m}_{opt}$ that generates the lowest BER among all design solutions produced in Fig. 5b-II. In the first loop of this iterative

process, the design $\mathbf{m}_{opt}$ with the lowest filter complexity M_c is selected (Fig. 5b-III) and its corresponding kernel coefficients h_p are used in the synthetization of the model S, realized through Bayesian-based ILA (at $Max_{it} = 50$ iterations), as shown in Fig. 5b-IV. For the design of the DPD filter, we assumed a Volterra-based architecture. Subsequently, the obtained DPD filter was used to pre-distort the test signal that unlike the probe signal was generated with a RRC pulse-shape (roll-off factor of 0.1), for which two modulation formats (DP-64QAM or DP-256QAM) and two symbol rates (64 and 80 GBd) could be selected. After the transmission and reception of the pre-distorted test signal (Fig. 5b-V), the BER was measured by counting the errors in 1 million bits per measurement point. This measurement was realized at a fixed optical signal-to-noise ratio (OSNR), where a noise-loading stage consisting of an Erbium-doped fiber amplifier (EDFA) and a variable optical attenuator (VOA) was used. In the following loop, a new $\mathbf{m}_{opt}$ with a higher filter complexity M_c is tested using this same described procedure. This process only stops when selecting a new $\mathbf{m}_{opt}$ degrades the BER concerning the value acquired in the previous loop (Fig. 5b-VI). This methodology is inspired by the automatic early stopping criterion [45] and can efficiently handle eventual filter overfitting caused by the inappropriate selection of M_c, as shown in the next sub-sections. In the Bayesian-based ILA, we used $\gamma = 15$.

C. Different Driver Amplifier (DA) Gains

The insertion loss of DP-IQ modulators limits the available OSNR at the output of the transmitter. To counter-balance this problem, increasing the drive voltage of the DP-IQ modulator (i.e., increasing the DA gain) is a common approach to achieve a higher OSNR. However, this provokes distortions due to DA saturation and the nonlinear response of the DP IQ-modulator, consequently, leading to different optimal DPD filter designs with different optimal memory tap distribution. In order to verify that the proposed approach is able to adapt the DPD design to the system configuration, we tested three gain configurations (g_1, g_2 and g_3) of the DAs to excite different degrees of transmitter nonlinearity. These gain configurations correspond to three levels of nonlinear system excitation: weak (g_1), strong (g_2) and highly nonlinear (g_3), such that $g_1 < g_2 < g_3$. For each DA gain configuration, we performed the validation methodology depicted by Fig. 5b.

As previously mentioned, during the SI, different optimal designs $\mathbf{m}_{opt}$ with different M_c were selected and stored to be tested in the performance evaluation as means to avoid overfitting and identify an optimal M_c. In order to pre-select these designs, the following policy was applied. First, the Bayesian-based SI is performed for the interval $M_C \, \varepsilon$ [50, 1400] with steps $\Delta M_C = 50$, what can be visually inspected from the identification error curves in Fig. 6 (solid dot markers). Second, to avoid an exhaustive testing of all these data points, only candidate solutions between $M_C = 50$ and $M_C = 1400$ at the elbows of the *plateaus* of the curves (indicated by the triangular markers in Fig. 6) were sampled. These sampled points represent

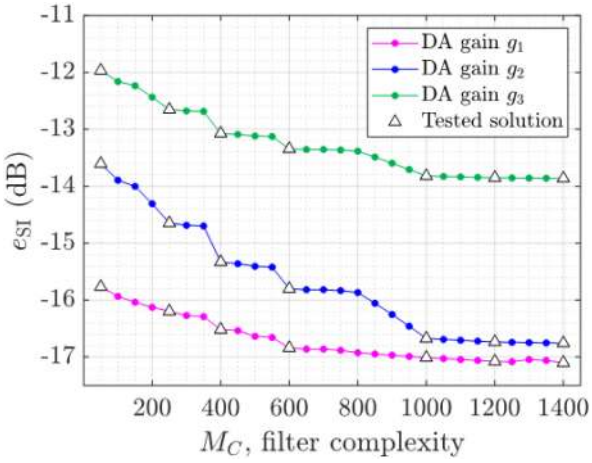

Fig. 6. Identification error for different DA gains with respect to filter complexity M_C when Bayesian-based SI is performed. Only the designs denoted by the triangular markers were tested in the Bayesian-based ILA.

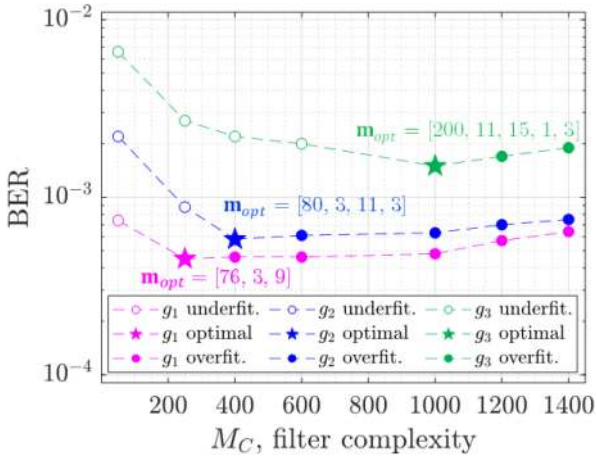

Fig. 7. BER validation (at fixed OSNR = 44.9 dB) for different DA gains ($g_1 < g_2 < g_3$) as function of the filter complexity M_C. The pre-selected solutions (Fig. 6) are validated, where it is possible to distinguish an optimum among underfitted and overfitted filter designs.

local solutions where diminishing e_{SI} is no longer worth the additional cost of increasing M_c.

The performance evaluation was carried out with a test signal configured to DP-64QAM at 64 GBd at maximum OSNR (44.9 dB). As can be seen in Fig. 7, the filter architecture that results in lowest BER for the gain g_1 (indicated by the pink star-like marker) is a 3rd-order Volterra filter ($\mathbf{m}_{opt}$ = [76, 3, 9], $M_C = 50$). When gain g_2 is tested, not only m_1 and m_2 incorporate 4 and 2 more taps, respectively, but there is also the inclusion of a 4th order ($\mathbf{m}_{opt}$ = [80, 3, 11, 3], $M_C = 400$). Finally, for the highly nonlinear regime, represented by the gain g_3, the design for lowest BER is a 5th-order filter ($\mathbf{m}_{opt}$ = [200, 11, 15, 1, 3], $M_C = 1000$). Fig. 7 also shows that a further increase of the filter complexity beyond the early stopping criterion only degrades the system performance due to model overfitting (solid dot markers).

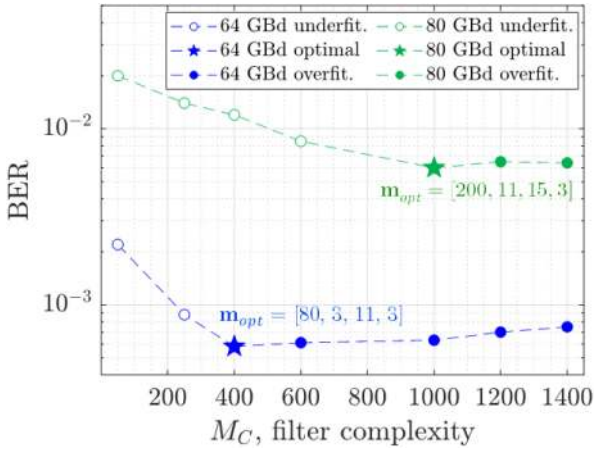

Fig. 8. BER validation (at fixed OSNR = 44.9 dB) for different symbol rates (64 and 80 GBd) as function of the filter complexity M_C.

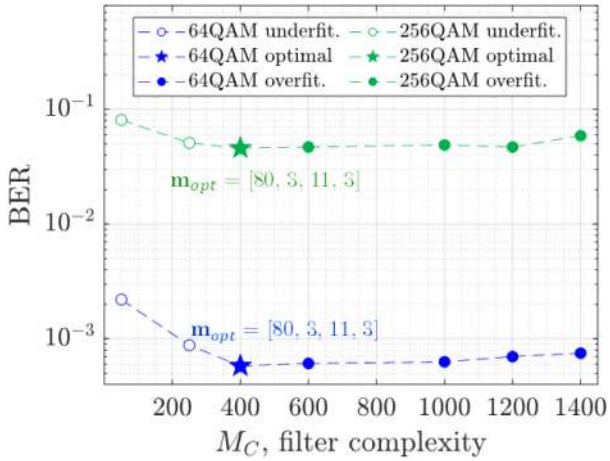

Fig. 9. BER validation (at fixed OSNR = 44.9 dB) for modulation formats (DP-64QAM and DP-256QAM) as function of the filter complexity M_C.

D. Different Symbol Rates

As previously mentioned, during the SI a probe signal at 96 GBd was used to excite the SUT. This enables to synthesize a model R that covers sufficient frequency components to ensure that the test signals (at 64 and 80 GBd) will not be cut off in the frequency domain when applied to the inverse model S. The downside of such procedure is that the 96 GBd signal also excites transmitter nonlinearities at frequencies where the 64 or 80 GBd waveforms have no spectral support. Consequently, it is expected that at higher symbol rates the output signal of the SUT will manifest stronger distortions. The next logical step is to know whether our proposed scheme can adapt the filter design to reflect the necessity of additional or fewer memory taps to model the distortions induced in different symbol rate regimes. With that being said, we apply the validation process depicted in Fig. 5b to tune the DPD hyperparameters. For this test, the SI was performed in an similar way with respect to the analysis presented in sub-section C, except for the fact that now a fixed DA gain (g_2) was set.

In the performance evaluation, the test signal was configured to a fixed modulation format (DP-64QAM) while the chosen symbol rates were either set to 64 GBd or 80 GBd. As depicted in Fig. 8, the higher symbol rate regime (80 GBd) requires a filter complexity ($\mathbf{m}_{opt}$ = [200, 11, 15, 3], M_c = 1000) 2.5× higher than the 64 GBd case ($\mathbf{m}_{opt}$ = [80, 3, 11, 3], M_c = 400) to achieve the lowest BER. This demonstrates that the proposed approach can tailor specific filter designs for operation in different symbol rates, offering more means for self-configuration of CT.

E. Different Modulation Formats

Finally, an analysis was carried out to investigate the influence of different modulation formats. At this stage, we repeated the procedure presented in D but with a slightly different setup, i.e., in the performance evaluation we fixed symbol rate at 64 GBd and used two modulation format options, DP-64QAM and DP-256QAM. As indicated in the Fig. 9, with the same filter architecture it is possible to reach a minimum BER for both cases, suggesting that even for high cardinalities (DP-256QAM) compact but efficient filter architectures can be beneficial in the design of DPD schemes.

F. Performance Comparison

In this sub-section, we investigate the performance of the results of the DPD HT that yield the lowest BER for different symbol rate regimes, which are depicted by the star-like markers in Fig. 8. First, we demonstrate the impact of Bayesian-based ILA in comparison to using default hyperparameters. Second, we experimentally benchmark the Bayesian-based approach with linear DPD and the nonlinear DPD using the heuristic SI approach introduced in [10]. Finally, we demonstrate the gains in reducing filter complexity by using MP instead of Volterra series in the design of the DPD filter. To recapitulate, in all the three aforementioned scenarios the setup was configured to DA gain g_2 and the analysis was carried out at a fixed modulation format at DP-64QAM, while varying the symbol rates to either 64 or 80 GBd. It is also important to remind to the reader that the pre-distortion experiments performed in this sub-section follow the same methodology presented in section V.B, i.e., Bayesian-based SI is used to optimize the memory vector ($\mathbf{m}$) and this resulting architecture is applied to synthesize the model S in the Bayesian-based ILA, where the ILA loop parameters are tuned. At the end, a noise-loading test was realized.

1) Bayesian-Based ILA vs. Default Parameters: Using default ILA hyperparameters (λ, δ and G) is a common strategy for synthetization of DPD filter because it is fast, i.e., does not require any optimizations, and many times these off-the-shelf solutions already yield very good results. As previously described, one frequent engineering rule is to select λ close to, but not equal to 1, δ approximately 100-1000 times the variance of the input sequence and $G \leq 1$. To illustrate this procedure, by choosing $\lambda = 0.9999999$, $\delta = 900$ and $G = 0.9$, it is possible to reduce the required OSNR of Volterra DPD with respect to their corresponding linear DPD filters (extracting only m_1 from $\mathbf{m}$), by approximately 0.4 dB (at BER = 10^{-2}) for 64 GBd and $+\infty$ for 80 GBd, as indicated in Fig. 10a. When the Bayesian-based

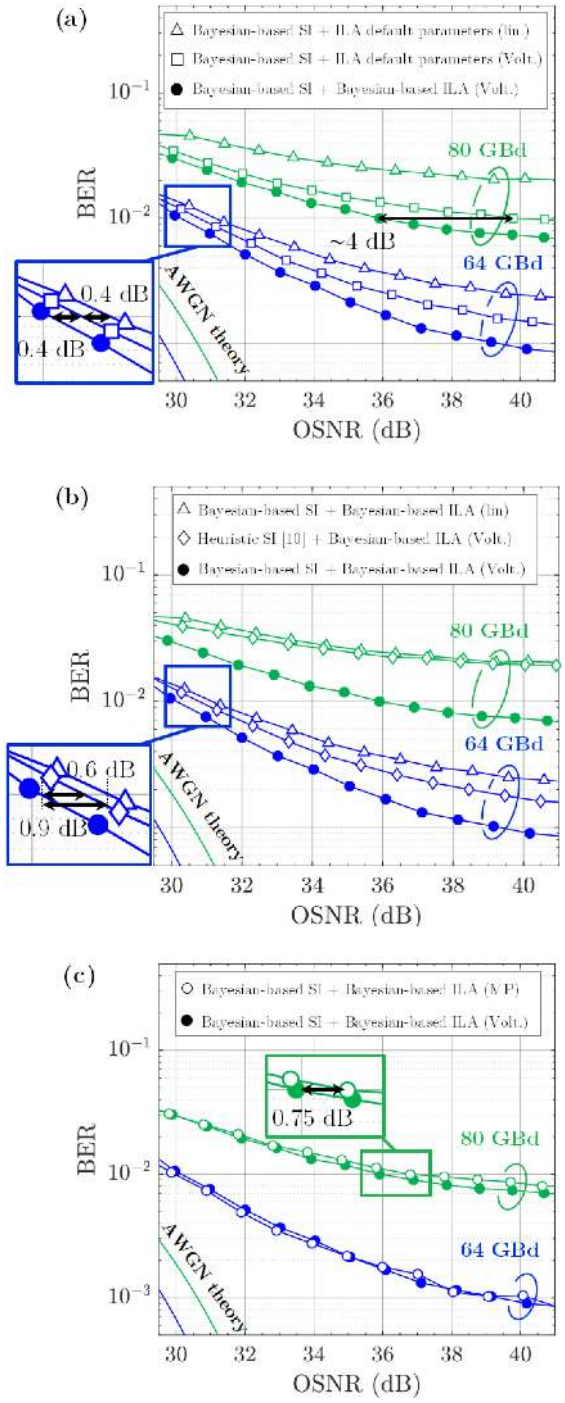

Fig. 10. BER curves obtained for DP-64QAM at 64 GBd and 80 GBd (a) when Bayesian-based ILA is benchmarked with the use of default parameters ($\lambda = 0.9999999$, $\delta = 900$ and $G = 0.9$), (b) when Bayesian-based SI is benchmarked with the heuristic SI approach proposed in [10] and (c) when Volterra and MP filters are used as DPD filters.

ILA is applied to optimize λ, δ and G, instead of using default values, it is possible to see a slight improvement in the reduction of the required OSNR of approximately 0.4 dB for 64 GBd (at BER $= 10^{-2}$). This improvement becomes more significant when the analysis is carried out for 80 GBd, where a 4-dB reduction is achieved, as depicted in Fig. 10a. This demonstrates the impact

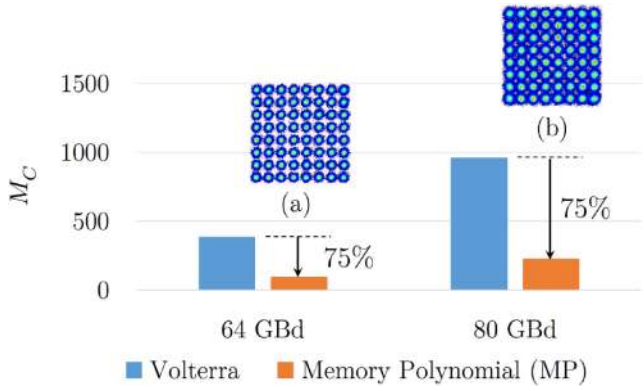

Fig. 11. Filter complexity reduction of 75% can be achieved by using MP pre-distorting filters. (a) 64 and (b) 80 GBd constellations at BER $= 6.6\text{x}10^{-4}$ and $5.9\text{x}10^{-3}$ (OSNR $= 44.9$ dB) respectively, when MP are used.

that the HT has in the synthetization of the DPD filter designs when compared to fixed ready-to-use hyperparameters.

2) Bayesian-Based SI vs. Grid-Based Heuristic Approach: In sub-section A, we showed with a simulative scenario that the Bayesian-based SI reaches an identification error floor requiring with less processing time than the grid-based heuristic approach introduced in [10]. Here, we experimentally show the optical performance of both techniques by restricting the number of iterations ($Max_{it} = 50$ iterations).

As can be seen in Fig. 10b, using Bayesian-based DPD for 64 GBd reduces the required OSNR by approximately 0.9 dB with respect to the linear DPD (at BER $= 10^{-2}$) and 0.6 dB in comparison to the benchmarked SI method. For the 80 GBd case, the benefit of using our proposed scheme becomes even more relevant, given that both benchmarks cannot operate below BER $= 10^{-2}$, whereas the Bayesian-based approach achieves a BER below this threshold.

3) DPD Filter via Volterra vs. MP: Finally, we test the performance of the proposed scheme with the use of MP pre-distorting (PD) filters. MPs represent a very compact subset of the Volterra series and mathematically correspond to rewriting Eq. 6, such that $h_p\ [c_1, \ldots, c_p] = 0,\ \forall\ c_1 \neq c_2 \neq \ldots \neq c_p$, which in simple words is equivalent to only considering the main diagonal of the p-th order Volterra operator h_p. By employing MP, the complexity of the pre-distortion filter drastically drops enabling simpler architectures. However, the question to answer is whether using this filter architecture could lead to a loss of performance. In order to answer this question, we performed the Bayesian-based ILA using MP as a filter prototype for the inverse model S and compared it with the use of Volterra filters. As can be seen in Fig. 10c, the incorporation of MP into the Bayesian-based ILA leads to a negligible loss of performance with respect to Volterra PD filters for the 64 GBd regime. Moreover, the filter complexity (M_c) reduces by approximately 75%, as depicted in Fig. 11. For the 80 GBd case, the performance is almost identical, with MP with a slightly higher required OSNR, perceived for the high-OSNR regime (> 34 dB). This can be explained by the fact that for such high symbol rate (80 GBd) the nonlinear distortions, represented by the kernel coefficients outside the main diagonal

of the Volterra filter ($h_p[c_1, \ldots, c_p] = 0, \forall\, c_1 \neq c_2 \neq \ldots \neq c_p$) become more relevant and adding these cross-terms, at the price of increasing filter complexity, can lead to a small improvement (0.75 dB at BER $= 10^{-2}$, Fig. 10c). In terms of filter complexity, a reduction of 75% is likewise achieved by the MP architecture (Fig. 11).

VI. Conclusion

In this work, we have investigated and experimentally demonstrated the benefits of employing Bayesian optimization, in Volterra-based system identification and digital pre-distortion of optical transmitters using the indirect learning architecture. The proposed approach offers means to provide efficient and automatic calibration of optical transmitters, which is a key requirement for the development of cognitive networks.

In Section II, we initially provided a comprehensive review on Bayesian optimization, which enabled the understanding, in Section IV, of how this optimization tool was incorporated to automatically tune important hyperparameters in the system identification (number of orders and memory tap distribution of the Volterra filter) and digital pre-distortion (indirect learning architecture loop parameters).

To evaluate the benefits of using the Bayesian optimization in the system identification, we numerically compared the proposed technique in Section V with a heuristic approach, where a 46% reduction in convergence time to reach a minimum identification error is achieved. Still in Section V, we introduced a method that can be used to validate the hyperparemeters that minimize bit error ratio, whilst avoiding model overfitting. This method was evaluated under three setup configuration scenarios, i.e., different driver amplifier gains, different symbol rates and different modulation formats. In this analysis, it was verified that by increasing the driver amplifier gain or signal symbol rate, stronger transmitter distortions are induced, thus, demanding more complex filter designs.

Finally, we assess the performance of the technique for three digital pre-distortion filter scenarios. First, demonstrating that the proposed approach improves the performance in comparison to using off-the-shelf indirect learning architecture hyperparemeters by reducing optical-to-noise ratio in 0.4 dB for a 64 GBd DP-64QAM signal. Second, it was inspected that the proposed technique also outperforms linear pre-distortion in 0.9 dB and the heuristic SI approach for nonlinear pre-distortion in 0.6 dB for 64 GBd DP-64QAM. Third, in conjunction with the use of memory polynomials a filter complexity reduction of approximately 75% with respect to Volterra filters is obtained while maintaining comparable performance.

A possible opportunity for future investigation in this topic is the development of a single optimization scheme to jointly tune the DPD filter architecture along with the ILA loop parameters to provide the best full forward performance.

References

[1] A. Napoli *et al.*, "Next generation elastic optical networks: The vision of the european research project IDEALIST," *IEEE Commun. Mag.*, vol. 53, no. 2, pp. 152–162, Feb. 2015.

[2] I. Miguel *et al.*, "Cognitive dynamic optical networks [Invited]," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 5, no. 10, pp. 107–118, Oct. 2013.

[3] B. Spinnler *et al.*, "Autonomous intelligent transponder enabling adaptive network optimization in a live network field trial," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 11, no. 9, pp. C1–C9, Jul. 2019.

[4] X. Dai, X. Li, M. Luo, and S. Yu, "Numerical simulation and experimental demonstration of accurate machine learning aided IQ time-skew and power-imbalance identification for coherent transmitters," *OSA Opt. Exp.*, vol. 27, no. 26, pp. 38367–38381, Dec. 2019.

[5] P. W. Berenguer, T. Rahman, A. Napoli, M. Nölle, C. Schubert, and J. K. Fischer, "Nonlinear digital pre-distortion of transmitter components," in *Proc. Eur. Conf. Exhib. Opt. Commun.*, Valencia, Spain, Sep. 2015, Paper Th.2.6.3.

[6] G. Khanna, B. Spinnler, S. Calabrò, E. De Man, and N. Hanik, "A robust adaptive pre-distortion method for optical communication transmitters," *IEEE Photon. Technol. Lett.*, vol. 28, no. 7, pp. 752–755, Apr. 2016.

[7] P. W. Berenguer, T. Rahman, A. Napoli, M. Nölle, C. Schubert, and J. K. Fischer, "Nonlinear digital pre-distortion of transmitter components," *IEEE/OSA J. Lightw. Technol.*, vol. 34, no. 8, pp. 1739–1745, Apr. 2016.

[8] C. Eun and E. J. Powers, "A new volterra predistorter based on the indirect learning architecture," *IEEE Trans. Signal Process.*, vol. 45, no. 1, pp. 223–227, Jan. 1997.

[9] M. Schetzen, *The Volterra and Wiener Theories of Nonlinear Systems*. Malabar, FL, USA: John Wiley Sons, 1980.

[10] A. Richter, S. Dris, and N. André, "On the analysis and emulation of nonlinear component characteristics," in *Proc. Opt. Fiber Commun. Conf. Exhib.*, San Diego, CA, USA, 2019, Paper Th.1.D.1.

[11] F. Musumeci *et al.*, "An overview on application of machine learning techniques in optical networks," *IEEE Commun. Surv. Tut.*, vol. 21, no. 2, pp. 1383–1408, Nov. 2018.

[12] C. Häger and H. D. Pfister, "Nonlinear interference mitigation via deep neural networks," in *Proc. Opt. Fiber Commun. Conf. Expo.*, San Diego, CA, USA, 2018, Paper W.3.A.4.

[13] T. T. Joy, S. Rana, S. Gupta, and S. Venkatesh, "Hyperparameter tuning for big data using bayesian optimisation," in *Proc. Int. Conf. Pattern Recognit.*, Cancun, Mexico, Dec. 2016, pp. 2574–2579.

[14] J. Mockus, V. Tiesis, and A. Zilinskas, *The Application of Bayesian Methods For Seeking the Extremum*. New York, NY, USA: North Holland, 1978.

[15] M. Sena *et al.*, "An autonomous identification and Pre-distortion scheme for cognitive transceivers using bayesian optimization," in *Proc. Eur. Conf. Exhib. Opt. Commun.*, Brussels, Belgium, Dec. 2020, Paper Tu.1.D.7.

[16] G. Khanna *et al.*, "A memory polynomial based digital pre-distorter for high power transmitter components," in *Proc. Opt. Fiber Commun. Conf. Expo.*, Los Angeles, CA, USA, 2017, Paper M.2.C.4.

[17] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.

[18] A. Bernardini, M. Carrarini, and S. De Fina, "The use of a neural net for copeing with nonlinear distortions," in *Proc. 20th Eur. Microw. Conf.*, Budapest, Hungary, 1990, pp. 1718–1723.

[19] B. E. Watkins and R. North, "Predistortion of nonlinear amplifiers using neural networks," in *Proc. Mil. Commun. Conf.*, McLean, VA, USA, 1996, pp. 316–320.

[20] R. Parisi, E. D. Di Claudio, G. Orlandi, and B. D. Rao, "Fast adaptive digital equalization by recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2731–2739, Nov. 1997.

[21] C. You and D. Hong, "Nonlinear blind equalization schemes using complex-valued multilayer feedforward neural networks," *IEEE Trans. Neural Netw.*, vol. 9, no. 6, pp. 1442–1455, Nov. 1998.

[22] E. Giacoumidis *et al.*, "Fiber nonlinearity-induced penalty reduction in CO-OFDM by ANN-based nonlinear equalization," *OSA Opt. Lett.*, vol. 40, no. 21, pp. 5113–5116, 2015.

[23] J. Zhang, M. Gao, W. Chen, and G. Shen, "Non-data-aided k-nearest neighbors technique for optical fiber nonlinearity mitigation," *IEEE/OSA J. Lightw. Technol.*, vol. 36, no. 17, pp. 3564–3572, May 2018.

[24] P. J. Freire *et al.*, "Complex-valued neural network design for mitigation of signal distortions in optical links," *IEEE/OSA J. Lightw. Technol.*, vol. 39, no. 6, pp. 1696–1705, Mar. 2021.

[25] J. Thrane, J. Wass, M. Piels, J. C. M. Diniz, R. Jones, and D. Zibar, "Machine learning techniques for optical performance monitoring from directly detected PDM-QAM signals," *IEEE/OSA J. Lightw. Technol.*, vol. 35, no. 4, pp. 868–875, Feb. 2017.

[26] W. S. Saif, M. A. Esmail, A. M. Ragheb, T. A. Alshawi, and S. A. Alshebeili, "Machine learning techniques for optical performance monitoring and modulation format identification: A survey," *IEEE Commun. Surv. Tut.*, vol. 22, no. 4, pp. 2839–2882, Sep. 2020.

[27] Y. Zhang, J. Xin, X. Li, and S. Huang, "Overview on routing and resource allocation based machine learning in optical networks," *Elsevier Opt. Fiber Technol.*, vol. 60, Dec. 2020, Art. no. 102355.

[28] V. Bajaj, F. Buchali, M. Chagnon, S. Wahls, and V. Aref, "Single-channel 1.61 Tb/s optical coherent transmission enabled by neural network-based digital pre-distortion," in *Proc. Eur. Conf. Exhib. Opt. Commun.*, Brussels, Belgium, 2020, Paper. Tu.1.D.5.

[29] G. Paryanti, H. Faig, L. Rokach, and D. Sadot, "A direct learning approach for neural network based pre-distortion for coherent nonlinear optical transmitter," *IEEE/OSA J. Lightw. Technol.*, vol. 38, no. 15, pp. 3883–3896, Aug. 2020.

[30] T. A. Eriksson, H. Bülow, and A. Leven, "Applying neural networks in optical communication systems: Possible pitfalls," *IEEE Photon. Technol. Lett.*, vol. 29, no. 23, pp. 2091–2094, Dec. 2017.

[31] M. Schaedler, M. Kuschnerov, S. Calabro, F. Pittala, C. Bluemm, and S. Pachnicke, "AI-based digital predistortion for IQ Mach-Zehnder modulators," in *Proc. Asia Comm. Photon. Conf.*, Chengdu, China, Nov. 2019, Paper S3B.3.

[32] T. Sasai *et al.*, "Wiener-Hammerstein model and its learning for nonlinear digital pre-distortion of optical transmitters," *OSA Opt. Exp.*, vol. 28, no. 21, pp. 30952–30963, 2020.

[33] M. Tzur, G. Paryanti, and D. Sadot, "Optimization of recurrent neural network-based pre-distorter for coherent optical transmitter via stochastic orthogonal decomposition," in *Proc. OSA Adv. Photon. Congr.*, 2020, Paper SpTh2I.5.

[34] Y.-Y. Lin *et al.*, "Reduction in Complexity of volterra filter by employing $\ell 0$-Regularization in 112-Gbps PAM-4 VCSEL optical interconnect," in *Proc. Opt. Fiber Commun. Conf. Expo.*, San Diego, CA, USA, 2020, Paper Th2A.51.

[35] P. Neary, "Automatic hyperparameter tuning in deep convolutional neural networks using asynchronous reinforcement learning," in *Proc. Int. Conf. Cogn. Comput.*, San Francisco, CA, USA, Sep. 2018, pp. 73–77.

[36] R. D. Nowak, "Penalized least squares estimation of volterra filters and higher order statistics," *IEEE Trans. Signal Process.*, vol. 46, no. 2, pp. 419–428, Feb. 1998.

[37] P. Händel, "Understanding normalized mean squared error in power amplifier linearization," *IEEE Microw. Wireless Compon. Lett.*, vol. 28, no. 11, pp. 1047–1049, Oct. 2018.

[38] V. J. Mathews and G. L. Sicuranza, *Polynomial Signal Process.* New York, NY, USA: John Wiley Sons, 2000.

[39] M. Hayes, *Statistical Digital Signal Processing and Modeling*. New York, NY, USA: John Wiley Sons, 1996.

[40] M. Nölle, M. S. Erkılınc, R. Emmerich, C. Schmidt-Langhorst, R. Elschner, and C. Schubert, "Characterization and linearization of high bandwidth integrated optical transmitter modules," in *Proc. Eur. Conf. Exhib. Opt. Commun.*, Dec. 2020, Paper Tu2D-4.

[41] C. R. S. Fludger and T. Kupfer, "Transmitter impairment mitigation and monitoring for high baud-rate, high order modulation systems," in *Proc. Eur. Conf. Exhib. Opt. Commun.*, Dusseldorf, Germany, 2016, pp. 1–3.

[42] B. Szafraniec, B. Nebendahl, and T. Marshall, "Polarization demultiplexing in stokes space," *OSA Opt. Exp.*, vol. 18, no. 17, pp. 17928–17939, Aug. 2010.

[43] M. Selmi, Y. Jaouen, and P. Ciblat, "Accurate digital frequency offset estimator for coherent polmux QAM transmission systems," in *Proc. 35th Eur. Conf. Opt. Commun.*, Vienna, Austria, 2009, pp. 1–2.

[44] T. Pfau, S. Hoffmann, and R. Noe, "Hardware-efficient coherent digital receiver concept with feedforward carrier recovery for M-QAM constellations," *IEEE/OSA J. Lightw. Technol.,* vol. 27, no. 8, pp. 989–999, Apr. 2009.

[45] L. Prechelt, "Automatic early stopping using cross validation: Quantifying the criteria," *Elsevier Neural Netw.,* vol. 11, no. 4, pp. 761–767, Jun. 1998.