Kernel mapping for mitigating nonlinear impairments in optical short-reach communications

LU ZHANG,[1*] XIAODAN PANG,[5] ALEKSEJS UDALCOVS,[3] OSKARS OZOLINS,[3] RUI LIN,[2] XIN YIN,[6] MING TANG,[7] WEIJUN TONG,[8] SHILIN XIAO,[4] AND JIAJIA CHEN[2*]

[1]College of Information Science and EE, Zhejiang University, Hangzhou 310027, China
[2]Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, Sweden
[3]NETLAB, RISE AB, Kista 16425, Sweden
[4]State Key Laboratory of Advanced Optical Communication System and Networks, Shanghai Jiao Tong University (SJTU), Shanghai 200240, China
[5]Infinera, Stockholm 117 43, Sweden
[6]Ghent University - IMEC, IDLab, Department of Information Technology, Gent 9000, Belgium
[7]Wuhan National lab for Optoelectronics (WNLO), Huazhong University of Sci&Tech (HUST), Wuhan 430074, China
[8]Yangtze Optical Fibre and Cable Company Ltd (YOFC), Wuhan 430073, China
**zhanglu1993@zju.edu.cn; jiajiac@chalmers.se*

Abstract: Nonlinear impairments induced by the opto-electronic components are one of the fundamental performance-limiting factors in high-speed optical short-reach communications, significantly hindering capacity improvement. This paper proposes to employ a kernel mapping function to map the signals in a Hilbert space to its inner product in a reproducing kernel Hilbert space, which has been successfully demonstrated to mitigate nonlinear impairments in optical short-reach communication systems. The operation principle is derived. An intensity modulation/direct detection system with 1.5-µm vertical cavity surface emitting laser and 10-km 7-core fiber achieving 540.68-Gbps (net-rate 505.31-Gbps) has been carried out. The experimental results reveal that the kernel mapping based schemes are able to realize comparable transmission performance as the Volterra filtering scheme even with a high order.

© 2019 Optical Society of America under the terms of the OSA Open Access Publishing Agreement

1. Introduction

Driven by emergence of cloud applications, more than 70% of network traffic stays within the datacenters, where high-speed short-reach transmission techniques are highly demanded [1]. Optical fiber transmission has evolved over decades and been widely recognized as the most cost- and energy-efficient technique to offer ultra-high capacity [1, 2]. The datacenter I/O port transmission speed is soon approaching 200-Gbps per lambda [3]. To meet such a high data rate, the intensity modulation direct detection (IM/DD) systems with advanced modulation formats such as pulse amplitude modulation (PAM) or discrete multi-tone (DMT) have been experimentally demonstrated [4-8, 13]. However, such an IM/DD system imposes stringent requirements on optoelectronic devices, bringing a significant challenge to cost-sensitive datacenters. Meanwhile, spatial division multiplexing (SDM) techniques provide an alternative approach to cope with an ever-increasing capacity demand, showing a great potential to achieve 1-Tbps and beyond [9]. Novel system techniques, such as vertical cavity surface emitting laser (VCSEL) with multi-core fiber (MCF), are promising to increase the spatial bandwidth density [10]. Although the cost-effective broadband opto-electrical components have been developing rapidly to keep pace with the capacity requirement of optical short-reach systems, the impairments, especially the nonlinear ones, introduced by these components to the high-speed short-reach communication systems cannot be neglected and become a significant issue that hinders the capacity improvement [2]. These nonlinear impairments come from non-idealities of various opto-electronic devices [2, 11], such as the nonlinear modulation characteristics from the laser, saturated power amplification from the optical/electrical amplifier, square law detection of the signal at the photodetector, and consequently are difficult to be mitigated by using any analytical modeling.

Recently, Volterra filtering [12, 13] and machine learning (ML) algorithms [14, 15] have been introduced as numerical methods for channel equalization against the nonlinear impairments in 200-Gbps and beyond optical short-reach communication systems [4-8, 13]. Compared with the linear signal processing methods employed only in one-dimensional scalar signal space, such as feed forward equalization, these methods handle the signal recovery in a complete and sequential high-dimensional space, known as a Hilbert space (HS). However, they still have limited benefits to mitigate nonlinear impairments. The Volterra filtering scheme uses high-order multiplications of all the inputs that have been applied to the system, and its complexity increases exponentially with nonlinear kernel orders. Typically, to keep complexity in an acceptable level, the nonlinear kernel order of no more than 3 is implemented [12, 13]. However, a small order is limited to compensate the complicated nonlinear impairments in optical short-reach systems. Many ML algorithms, such as [14, 15], compensate the nonlinear impairments through learning from multiple neural layers, which also treat the signal in a high-dimensional manner. Nevertheless, the more the neural layers are introduced, the higher the computational complexity is. It remains an open question whether the high computational complexity of ML-based equalizers are proper in optical short-reach communications.

From the signal processing perspective, the aforementioned challenge of the existing methods that mitigate the nonlinear impairments is referred to as ‘curse of dimension’ dilemma [16]. When the signal dimension increases, the performance can be improved but causing intolerable complexity. It often happens that when increasing the signal dimension up to a certain level, a very little performance improvement needs a dramatic increase of the calculations. One way to address this dilemma is to find a suitable dimension mapping function. The kernel mapping, also referred to as ‘kernel trick’ [18], was proposed to address the ‘curse of dimension’, and hence has a great potential to tackle with the issues caused by complicated nonlinear impairments in optical short-reach communications as well. A kernel mapping function maps the signals in a HS to the inner product in a reproducing kernel Hilbert space (RKHS). Therefore, the complicated calculations caused by a high (or infinite)-dimensional space are not needed.

In this regard, this paper introduces the kernel mapping method for signal processing in optical short-reach communications, where ‘Mercer kernel’ is utilized to map the signals in the HS to the inner product in the RKHS. A 540.68-Gbps (net-rate 505.31-Gbps) optical short-reach system with IM/DD, DMT, 1.5-µm VCSEL and 10-km 7-core fiber is carried out to experimentally demonstrate the kernel mapping idea. In this paper, we significantly extend the previous work reported in [19] by: 1) providing theoretical analysis and operational principles of the kernel mapping; 2) introducing mathematical derivations of the kernel least mean square (KLMS) and kernel recursive least squares (KRLS) algorithms; 3) showing experiments results for the DSP flow optimization process; 4) extending the experimental results with both KLMS and KRLS algorithms. The results reveal that kernel mapping is effective to mitigate the nonlinear impairments for short-reach optical communications. The proposed KLMS and KRLS achieve comparable transmission performance with Volterra filtering scheme.

2. Operation principles

In this section, we introduce operational principles of the kernel mapping for nonlinear impairment mitigation in short-reach optical communication systems.

In this paper, matrices are denoted by boldface capital letters, vectors are denoted by boldface lowercase letters, and scalars are denoted by lowercase letters. A boldface lowercase letter with a subscript represents a vector in the vector space. A boldface lowercase letter with blankets means a vector sampled at a given moment, while a normal lowercase letter with blankets means a scalar at a given sampling point.

The basics of HS and RKHS are detailed in *Appendix 7.1* and *Appendix 7.2*, respectively. To employ kernel mapping in the optical communications, the signal processing in the RKHS is done with the inner production of the signals. To further improve the analytic power of the RKHS in optical communications, kernel mapping is needed for transforming the reproducing kernel κ ($\boldsymbol{x}_i$, $\boldsymbol{x}_j$) from the RKHS to the feature space F composed of feature vectors φ (.) of the RKHS.

According to Mercer's theorem, any reproducing kernel κ ($\boldsymbol{x}_i$, $\boldsymbol{x}_j$) in the RKHS can be expanded with the non-negative eigenvalues λ_p and eigenfunctions θ_p, which is expressed in (1).

$$\kappa(\boldsymbol{x_i},\boldsymbol{x_j})=\sum\nolimits_{\boldsymbol{p}=1}^{\infty}\lambda_p\theta_{\boldsymbol{p}}(\boldsymbol{x_i})\theta_{\boldsymbol{p}}(\boldsymbol{x_j}). \tag{1}$$

Then, kernel mapping φ (.) is constructed as follows:

$$\varphi(.)=\left[\sqrt{\lambda_1}\theta_1(.),\sqrt{\lambda_2}\theta_2(.),...\right]\big|H\to F\,, \tag{2}$$

where φ **(.)** is a kernel vector (i.e., a feature vector) in the feature space F. The dimension of the kernel mapping feature space is determined by the number of positive eigenvalues, which is infinite when Gaussian kernel function is employed. Thus, F is equal to RKHS with Gaussian kernel function. Without loss of generality, we do not distinguish F and RKHS in this paper. An important rule of the kernel function κ ($\boldsymbol{x}_i$, $\boldsymbol{x}_j$) is expressed as:

$$\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j)=\varphi(\boldsymbol{x}_i)^T\varphi(\boldsymbol{x}_j). \tag{3}$$

A concrete example of expanding calculation of 2 samples $x_{i,p}$ and $x_{i,q}$ in Gaussian kernel with (3) is presented as follows.

$$\begin{aligned}\kappa(x_{i,p},x_{i,q})&=\exp(-(x_{i,p}-x_{i,q})^2/2\sigma^2)\\&=\exp(-({x_{i,p}}^2+{x_{i,q}}^2)/2\sigma^2)\cdot\sum\nolimits_{m=0}^{\infty}\frac{(x_{i,p}x_{i,q})^m}{m!\sigma^m}\\&=\sum\nolimits_{m=0}^{\infty}\left\{\exp(-\frac{{x_{i,p}}^2}{2\sigma^2})\exp(-\frac{{x_{i,q}}^2}{2\sigma^2})\sqrt{\tfrac{1}{m!}}\sqrt{\tfrac{1}{m!}}\frac{{x_{i,p}}^m}{\sigma^{m/2}}\frac{{x_{i,q}}^m}{\sigma^{m/2}}\right\},\\&=\varphi(x_{i,p})^T\varphi(x_{i,q})\end{aligned} \tag{4a}$$

$$\varphi(x_{i,p})=\exp(-\frac{{x_{i,p}}^2}{2\sigma^2})(1,\sqrt{\tfrac{1}{1!}}\frac{x_{i,p}}{\sigma},\sqrt{\tfrac{1}{2!}}\frac{{x_{i,p}}^2}{\sigma^2},...). \tag{4b}$$

Eq. (4) demonstrates the Gaussian kernel function is capable of processing all the orders of the kernels, i.e., (1, $\boldsymbol{x}$, $\boldsymbol{x}^2$, $\boldsymbol{x}^3$, …), where the higher orders ($\boldsymbol{x}^2$, $\boldsymbol{x}^3$, …) represent nonlinear distortions. Therefore, after the kernel mapping the nonlinear effects are transformed into the inner production. However, the usage of the Gaussian kernel function in our proposed method does not need the concrete expansion shown in (4b), while the inner production shown in (23) in *Appendix 7.2* is enough.

It is easy to find that the feature space F is the same as the RKHS since they share the same basis set. If we denote a vector $\boldsymbol{\rho}$ in the RKHS as $\boldsymbol{\rho}=\sum\nolimits_{\boldsymbol{i}=1}^{\boldsymbol{m}}a_i\boldsymbol{\varphi}(\boldsymbol{x_i})$, any continuous mapping $\{f\,|\,R^t\text{->}R\}$ in the RKHS meets (5) [18].

$$\left\|f-\boldsymbol{\rho}^{\boldsymbol{T}}\boldsymbol{\varphi}(.)\right\|<\boldsymbol{\varepsilon},\forall\boldsymbol{\varepsilon}>0. \tag{5}$$

Eq. (5) implies that the linear model in the RKHS has the universal approximation property [20]. Combining (28) in *Appendix 7.2* and (5), we can recover the signal impaired by the high-order nonlinear distortions, in the RKHS in a linear manner. Such signal recovery is based on the optimization of $\boldsymbol{\varphi}$ and $\boldsymbol{\rho}$ with training data sets. The kernel mapping in the RKHS is illustrated in Fig. 1. The initial signal vector $\boldsymbol{x}$ is distorted by the nonlinear impairments, which are difficult to be modeled by any analytic solutions in the HS. After kernel mapping $\boldsymbol{\varphi}$ (**.**), we can get an analytic solution in the RKHS with feature vectors $\boldsymbol{\varphi}$ ($\boldsymbol{x}$). Then, $\boldsymbol{\rho}^{\mathbf{T}}\boldsymbol{\varphi}$ based transformation (i.e. matrix rotation) is used to map the signals in the RKHS to R, which can be easily processed with linear compensating schemes for mitigating the nonlinear impairments.

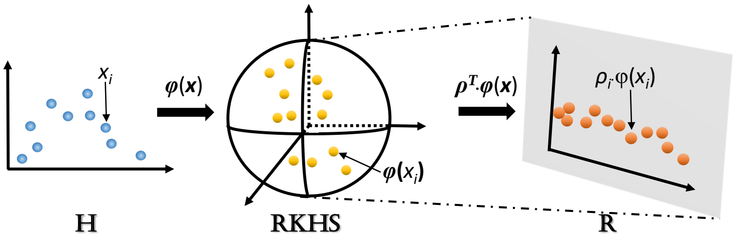
x_i
$\boldsymbol{\varphi}(\boldsymbol{x})$
$\boldsymbol{\varphi}(x_i)$
$\boldsymbol{\rho}^T\cdot\boldsymbol{\varphi}(\boldsymbol{x})$
$\rho_i\cdot\varphi(x_i)$
H
RKHS
R

Fig. 1. Schematic diagram of kernel mapping.

The strategy to employ kernel mapping is to formulate the classic linear signal processing in the RKHS to iteratively mitigate nonlinear impairments. We use the training data set $\{\boldsymbol{x}(k), d(k)\}|_{k=1,2,\ldots N}$, to obtain the optimal parameters for kernel mapping $\boldsymbol{\varphi}$ (**.**) as well as the equalization coefficients. Here, $\boldsymbol{x}(k)$ is the received signal vector at the k-th sampling point in the *HS*, $d(k)$ is the transmitted signal at the k-th sampling point, and N is the training data size. $\boldsymbol{w}(k)|_{k=1,2,\ldots N}$ is the estimated filter weight vector at the k-th sampling point. Assuming $\boldsymbol{d}$=(d(1), d(2), …, d(N)), the error signal as the difference between $\boldsymbol{d}$ and kernel mapped signal is shown as follow:

$$\boldsymbol{e} = \boldsymbol{d} - \sum_{k=1}^{N} \mathbf{w}(k)\varphi(\boldsymbol{x}(k)). \tag{6}$$

Eqs. (6) and (28) in *Appendix 7.2* are the same if (3) is satisfied. The geographical interpretation of the error signal $\boldsymbol{e}$, desired signal $\boldsymbol{d}$ and kernel mapped signal is shown in Fig. 2. The kernel mapping and the equalization coefficients can reach the optimization convergence point when the error signal is perpendicular to the kernel mapping space [17]. This principle shares with many linear signal processing methods. As a result, combing the kernel mapping and linear signal processing algorithms, such as LMS and RLS, makes optical short-reach communication systems tolerant to nonlinear impairments.

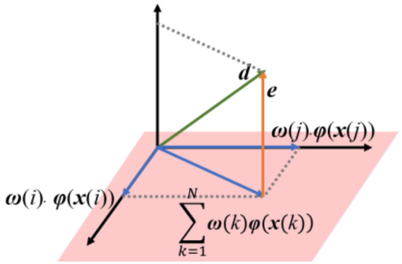
$\boldsymbol{d}$
$\boldsymbol{e}$
$\boldsymbol{\omega}(j)\cdot\boldsymbol{\varphi}(\boldsymbol{x}(j))$
$\boldsymbol{\omega}(i)\cdot\ \boldsymbol{\varphi}(\boldsymbol{x}(i))$
$\sum_{k=1}^{N}\boldsymbol{\omega}(k)\boldsymbol{\varphi}(\boldsymbol{x}(k))$

Fig. 2. Geographic interpretation of the signal processing space with kernel mapping.

2.1 Kernel least mean square (KLMS) algorithm

The design of the KLMS algorithm at the k-th time sampling point in the RKHS is inspired by the classic LMS algorithm, which is expressed as:

$$e(k) = d(k) - \boldsymbol{w}(k-1)^{T}\varphi(\boldsymbol{x}(k)), \tag{7a}$$

$$\boldsymbol{w}(k) = \boldsymbol{w}(k-1) + \eta e(k)\varphi(\boldsymbol{x}(k)), \tag{7b}$$

where $e(k)$ is the error signal between transmitted signal $d(k)$ and equalized signal, $\boldsymbol{w}(k)$ is the estimated filter weight vector, and η is the step-size parameter. By calculating the weight updating formula in (7b) iteratively, we get:

$$\begin{aligned}\boldsymbol{w}(k) &= \boldsymbol{w}(k\text{-}1) + \eta e(k)\varphi(\boldsymbol{x}(k)) \\ &= \boldsymbol{w}(k\text{-}2) + \eta(e(k\text{-}1)\varphi(\boldsymbol{x}(k\text{-}1) + e(k)\varphi(\boldsymbol{x}(k))) = \ldots \\ &= \eta\sum_{\boldsymbol{i}=1}^{\boldsymbol{k}} e(i)\varphi(\boldsymbol{x}(i)) \quad (\boldsymbol{w}(0) = 0).\end{aligned} \tag{8}$$

Thus, after k-th training samples, the weight estimation is expressed as a linear combination of all the previous and present transformed inputs, weighted by the prediction errors (and scaled by η). This also matches the theorem employing $\boldsymbol{\rho}$ in (5) [20]. When the estimated weights are used to compensate a new set of received vector $\boldsymbol{x}$', the recovered signal is calculated by:

$$\boldsymbol{w}(k)^{\boldsymbol{T}}\varphi(\boldsymbol{x}') = (\eta\sum_{\boldsymbol{i}=1}^{\boldsymbol{k}} e(i)\varphi(\boldsymbol{x}(i)))\varphi(\boldsymbol{x}') = \eta\sum_{\boldsymbol{i}=1}^{\boldsymbol{k}} e(i)\kappa(\boldsymbol{x}(i), \boldsymbol{x}'). \tag{9}$$

It is interesting to find the weights $\boldsymbol{w}$ absent in the equalized signals that are influenced by the nonlinear impairments. Instead, the sum of all past errors is multiplied by the kernel mapping on the previously received data (i.e., training data). It proves that the kernel mapping is a process without referring to the coefficient $\boldsymbol{w}$ and the high-dimensional feature vector $\boldsymbol{\varphi}$. Thus, the 'kernel trick' proposed for optical communications can well address the nonlinear impairments in a high-dimensional manner and avoid the 'curse of dimension' efficiently.

Then, the KLMS algorithm can be expressed as follows.

$$f_{\boldsymbol{k}-1}(\boldsymbol{x}(k)) = \eta\sum_{\boldsymbol{i}=1}^{\boldsymbol{k}-1} e(i)\kappa(\boldsymbol{x}(i), \boldsymbol{x}(k)), \tag{10a}$$

$$e(k) = d(k) - \boldsymbol{f}_{k-1}(\boldsymbol{x}(k)), \tag{10b}$$

$$f_k(\cdot) = f_{k-1}(\cdot) + \eta e(k)\kappa(\boldsymbol{x}(k), \cdot). \tag{10c}$$

The function f_k is the equalization mapping at the k-th iteration, which also corresponds to the k-th time sampling point. Similar as the LMS algorithm, (10a) is a mapping function, (10b) is an error function and (10c) is an updating function. The signal processing flow of KLMS is shown as in Algorithm 1. N_L is the number of iterations, also the training overhead. All the nonlinear equalizer coefficients are derived using a training data set.

Algorithm 1. The signal processing flow of the KLMS.

```
1.  Set $\eta$ and pre-defined threshold $e_t$;
2.  Choose kernel function $\kappa$, $k$=1;
3.  while $k \le N$ do
      i.     $f_{\boldsymbol{k}-1}(\boldsymbol{x}(k)) = \eta\sum_{\boldsymbol{i}=1}^{\boldsymbol{k}-1} e(i)\kappa(\boldsymbol{x}(i), \boldsymbol{x}(k))$
      ii.    $e(k) = d(k) - \boldsymbol{f}_{k-1}(\boldsymbol{x}(k))$
      iii.   save $x(k)$ and $e(k)$
      iv.    if $e(k) < e_t$
      v.         save $k$
      vi.        break
      vii.   else
      viii.      $k$=$k$+1
      ix.        continue
4.  end
5.  Training output: $N_L$=$k$, {$\boldsymbol{x}(k)$, $e(k)$, k=1,2…$N_L$ }.
6.  Equalization output:
    $f(\boldsymbol{x}') = \eta\sum_{\boldsymbol{k}=1}^{N_L} \boldsymbol{e}(\boldsymbol{k})\kappa(\boldsymbol{x}(\boldsymbol{k}), \boldsymbol{x}')$
```

2.2 Kernel recursive least squares (KRLS) algorithm

The KLMS deals with the instantaneous value of the squared estimation error, and hence faces the same convergence rate problem as that in the LMS. Comparatively, the RLS algorithm aims at minimizing the sum of the squared estimation errors, it always provides a convergence rate significantly faster and better performance than the LMS.

In the KRLS, by introducing:

$$\begin{cases} \boldsymbol{D}(k) = [d(1), d(2), ..., d(k)]^{\boldsymbol{T}} \\ \Phi(k) = [\varphi(\boldsymbol{x}(1)), \varphi(\boldsymbol{x}(2)), ..., \varphi(\boldsymbol{x}(k))]^{\boldsymbol{T}} \end{cases}, \tag{11}$$

we can get:

$$\boldsymbol{w}(k) = [\eta \boldsymbol{I} + \Phi(k)\Phi(k)^{\boldsymbol{T}}]^{-1}\Phi(k)\boldsymbol{D}(k). \tag{12}$$

Since **[A+BC]$^{-1}$B= B[A+CB]$^{-1}$**, (12) can be transformed to:

$$\boldsymbol{w}(k) = \Phi(k)[\eta \boldsymbol{I} + \Phi(k)^{\boldsymbol{T}}\Phi(k)]^{-1}\boldsymbol{D}(k). \tag{13}$$

Defining $\boldsymbol{P}(k)=[\eta\boldsymbol{I}+\Phi(k)^{\mathrm{T}}\,\Phi(k)]^{-1}$,

$$\boldsymbol{P}(k)^{-1} = \begin{bmatrix} \boldsymbol{P}(k-1)^{-1} & \Phi(k-1)^{\boldsymbol{T}}\varphi(\boldsymbol{x}(k)) \\ \varphi(\boldsymbol{x}(k))^{\boldsymbol{T}}\Phi(k-1) & \eta + \varphi(\boldsymbol{x}(k))^{\boldsymbol{T}}\varphi(\boldsymbol{x}(k)) \end{bmatrix}. \tag{14}$$

By introducing

$$\begin{cases} \boldsymbol{z}(k) = \boldsymbol{P}(k-1)\Phi(k-1)^{\boldsymbol{T}}\varphi(\boldsymbol{x}(k)) \\ \boldsymbol{r}(k) = \eta + \varphi(\boldsymbol{x}(k))^{\boldsymbol{T}}\varphi(\boldsymbol{x}(k)) - \boldsymbol{z}(k)^{\boldsymbol{T}}\Phi(k-1)\varphi(\boldsymbol{x}(k)) \end{cases}, \tag{15}$$

$\boldsymbol{P}(k)$ can be presented as:

$$\boldsymbol{P}(k) = \boldsymbol{r}(k)^{-1}\begin{bmatrix} \boldsymbol{P}(k-1)\boldsymbol{r}(k) + \boldsymbol{z}(k)\boldsymbol{z}(k)^{\boldsymbol{T}} & -\boldsymbol{z}(k) \\ -\boldsymbol{z}(k)^{\boldsymbol{T}} & 1 \end{bmatrix}. \tag{16}$$

Defining $\boldsymbol{\beta}(k)=\boldsymbol{P}(k)\boldsymbol{D}(k)$, we can get:

$$\begin{aligned} \boldsymbol{\beta}(k) &= \boldsymbol{r}(k)^{-1}\begin{bmatrix} \boldsymbol{P}(k-1)\boldsymbol{r}(k) + \boldsymbol{z}(k)\boldsymbol{z}(k)^{\boldsymbol{T}} & -\boldsymbol{z}(k) \\ -\boldsymbol{z}(k)^{\boldsymbol{T}} & 1 \end{bmatrix}\begin{bmatrix} \boldsymbol{D}(k-1) \\ \boldsymbol{x}(k) \end{bmatrix}. \\ &= \begin{bmatrix} \boldsymbol{\beta}(k-1) - \boldsymbol{z}(k)\boldsymbol{r}(k)^{-1}e(k) \\ \boldsymbol{r}(k)^{-1}e(k) \end{bmatrix} \end{aligned} \tag{17}$$

Here, $e(k)$ is the prediction error. The KRLS scheme at the k-th iteration is expressed as follows:

$$\begin{aligned} \boldsymbol{f}_{k-1}(\boldsymbol{x}(k)) &= \boldsymbol{\varphi}(\boldsymbol{x}(k))^{\boldsymbol{T}}\Phi(k-1)\boldsymbol{\beta}(k-1) \\ &= \sum_{\boldsymbol{i}}^{\boldsymbol{k}-1}\boldsymbol{\beta}_{\boldsymbol{i}}(k-1)\boldsymbol{\kappa}(\boldsymbol{x}(i), \boldsymbol{x}(k)), \end{aligned} \tag{18a}$$

$$e(k) = d(k) - \boldsymbol{f}_{k-1}(\boldsymbol{x}(k)), \tag{18b}$$

$$\boldsymbol{f}_k(\cdot) = \boldsymbol{f}_{k-1}(\cdot) + \boldsymbol{r}(k)^{-1}[\boldsymbol{\kappa}(\boldsymbol{x}(k), \cdot) - \sum_{\boldsymbol{i}=1}^{\boldsymbol{k}-1}\boldsymbol{z}_{\boldsymbol{i}}(k)\boldsymbol{\kappa}(\boldsymbol{x}(i), \cdot)]e(k). \tag{18c}$$

Similar as the RLS algorithm, (18a) is a mapping function, (18b) is an error function and (18c) is an updating function. The signal processing flow of the KRLS is shown as in Algorithm 2.

Algorithm 2. The signal processing flow of the KRLS.

```
Set $\eta$ and pre-defined threshold $\beta_t$;
Choose kernel function $\kappa$, $k$=1;
$\boldsymbol{P}(1)=(\eta+\kappa(\boldsymbol{x}(1),\boldsymbol{x}(1))^{-1}$;
$\boldsymbol{\beta}(1)=\boldsymbol{P}(1)\boldsymbol{D}(1)$;
while $k \le N$ do
    $\Phi(k)=[\varphi(\boldsymbol{x}(1)),\varphi(\boldsymbol{x}(2)),\ldots,\varphi(\boldsymbol{x}(k))]^{\boldsymbol{T}}$
    $\begin{cases} \boldsymbol{z}(k)=\boldsymbol{P}(k-1)\Phi(k-1)^{\boldsymbol{T}}\varphi(\boldsymbol{x}(k)) \\ \boldsymbol{r}(k)=\eta+\varphi(\boldsymbol{x}(k))^{\boldsymbol{T}}\varphi(\boldsymbol{x}(k))-\boldsymbol{z}(k)^{\boldsymbol{T}}\Phi(k-1)\varphi(\boldsymbol{x}(k)) \end{cases}$
    $\boldsymbol{P}(k)=\boldsymbol{r}(k)^{-1}\begin{bmatrix} \boldsymbol{P}(k-1)\boldsymbol{r}(k)+\boldsymbol{z}(k)\boldsymbol{z}(k)^{\boldsymbol{T}} & -\boldsymbol{z}(k) \\ -\boldsymbol{z}(k)^{\boldsymbol{T}} & 1 \end{bmatrix}$
    $\boldsymbol{f}_{k-1}(\boldsymbol{x}(k))=\sum_{i}^{k-1}\boldsymbol{\beta}_i(k-1)\boldsymbol{\kappa}(\boldsymbol{x}(i),\boldsymbol{x}(k))$
    $e(k)=d(k)-\boldsymbol{f}_{k-1}(\boldsymbol{x}(k))$
    $\boldsymbol{\beta}(k)=\begin{bmatrix} \boldsymbol{\beta}(k-1)-\boldsymbol{z}(k)\boldsymbol{r}(k)^{-1}e(k) \\ \boldsymbol{r}(k)^{-1}e(k) \end{bmatrix}$
    save $x(k)$ and $\beta(k)$
    if $\beta(k) < \beta_t$
        save $k$
        break
    else
        $k=k+1$
        continue
end
Training output: $N_L=k$, $\{\boldsymbol{x}(k), \boldsymbol{\beta}(k)$, k=1, 2…$N_L$ }.
Equalization output:
$f(\boldsymbol{x}')=\sum_{k=1}^{N_L}\boldsymbol{\beta}(k)\boldsymbol{\kappa}(\boldsymbol{x}(k),\boldsymbol{x}')$
```

3. Experimental setup

To quantify the benefits of using kernel trick on optical short-reach communication systems, a high-speed IM/DD system is experimentally carried out. The experimental setup is shown in Fig. 3, the digital signal processing (DSP) flow of DMT modulation and demodulation is also included in Fig. 3. The DMT signals are generated offline in MATLAB and loaded into a 92-GSa/s digital to analog converter (DAC). The length of the inverse fast Fourier transform (IFFT) points and cyclic prefix (CP) length of the DMT signal are set to 1024 and 16, respectively.

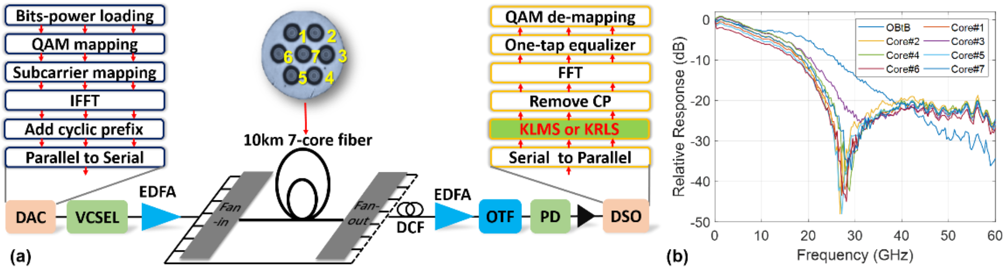

Fig. 3. (a) Experimental setup, and (b) measured system frequency response.

In the experiments, the VCSEL die is electrically driven via a 100-μm GSG 50-GHz probe. The light generated in the VCSEL is coupled into a single-mode fiber. There is no temperature controller (TEC) used in the setup. The VCSEL bias current is set to 7.8-mA. The measured central wavelength of the probed VCSEL is 1543.2-nm and the captured output optical power is 1-dBm. The maximum 3-dB bandwidth of VCSEL is about 22-GHz. The signal is split in a fan-in module of the 10-km 7-core multi-core fiber (MCF), using a 1:8 optical coupler. An Erbium-doped fiber amplifier (EDFA) with 14.8-dBm output power is used before the fan-in module to compensate for an extra loss from de-correlation. The optical delay lines are used to de-correlate the signals to emulate a practical system using seven independently modulated VCSELs. After the 10-km 7-core MCF, signals are detected individually after a fan-out module. The inter-core crosstalk is -45dB/100km, and therefore the cores of MCFs can be treated almost independently considering the typical reach within datacenters [21].

The 7 branches connected with 7 cores of MCF is connected to a DCF (-159 ps/nm) and the rest one is treated as the optical back-to-back (OBtB) transmission demonstration. The signal is amplified by a pre-amplifier EDFA and an optical tunable filter (OTF) is utilized to filter out the amplified spontaneous emission (ASE) noise. A 90-GHz PIN photo-detector (PD) is used at the receiver. A variable optical attenuator (VOA) is used before the PD. The signal after direct-detection is amplified by a 65-GHz linear electrical amplifier and captured by a 160-GSa/s digital storage oscilloscope (DSO). After DSO, the captured signal is processed offline.

The measured system frequency response with both OBtB transmission and MCF transmission is shown in Fig. 3(b). The response for the MCF based system is with DCF, which could reduce the influence of chromatic dispersion induced power fading and increase the system bandwidth. However, for short-reach optical communication systems where the fiber length is typically up to a few kilometers, the DCF is not always preferred because of extra deployment cost. In the OBtB case, 220 subcarriers are loaded. Here, bit-power loading [22] is used to improve the system capacity and spectrum efficiency. The quadrature amplitude modulation (QAM) orders vary from 64QAM to 16QAM are employed. In the 10-km MCF case where 210 subcarriers are loaded, bit-power loading is also employed.

The system nonlinear impairments mainly come from the VCSEL's chirp and its iterations with the fiber dispersion. Besides, some other factors, such as the high peak-to-average ratio of the DMT signal, inter-subcarrier mixing in square-law detection of photo detector, and saturation of the electrical amplifiers, also contribute to the nonlinearities of the short-reach optical link.

4. Experimental results and discussions

The kernel mapping is firstly used to optimize OBtB case, which is shown in Fig. 4. The original BER is measured with linear channel equalization. Here, Volterra filters considering the nonlinear distortions up to the 2nd-, 3rd- and 4th-order are compared. In the Volterra filtering, the tap numbers of the 2nd-, 3rd- and 4th-order nonlinear kernels are set to 15, 9 and 11, respectively. In the KLMS and the KRLS, the tap number is 3. With the 2nd-order Volterra filtering, the BER is reduced with an increased number of training overhead N_L and the BER is still higher than the continuously-interleaved Bose–Chaudhuri–Hocquenghem FEC (CI-BCH (1020, 988)) [23] with a BER-limit of 4.52×10^{-3} [24]. Volterra filtering reaches the CI-BCH FEC limit when the 3rd-order nonlinear distortions are considered. When the Volterra filtering keeps up to the 4th order, the BER performance improvement is limited but at the expense of much higher computational complexity. Thus, the Volterra filtering with nonlinear distortions up to the 3rd order is included in the following experimental results. In contrast, KLMS and KRLS reaches BER lower than the CI-BCH FEC limit with 4096 training samples. The 3rd-order Volterra filtering slightly outperforms the KLMS, and the KRLS outperforms the Volterra filtering.

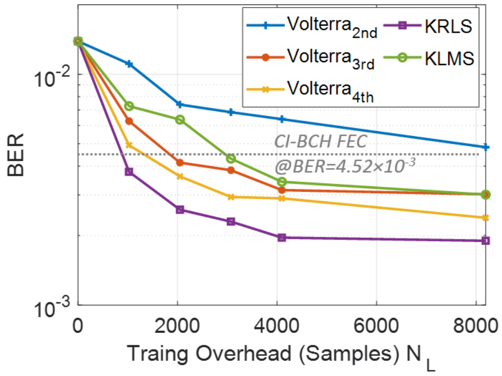

Fig. 4. The BER versus training overhead N_L

The BER performance versus the received optical power at OBtB case is shown in Fig. 5. The achieved line rate at OBtB is 91.72-Gbps. The linear equalization compensates the linear impairment and reduces the BER to ~ 3×10^{-2}. The system performance is greatly improved with Volterra, and the BER is reduced to CI-BCH FEC limit at 4.52×10^{-3}. The KLMS reaches the similar performance as Volterra filtering, and the KRLS outperforms the Volterra filtering.

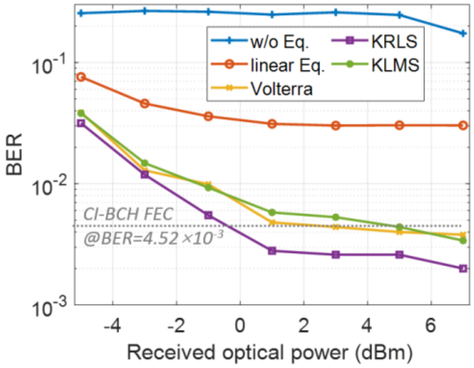

Fig. 5. BER for optical back-to-back case as a function of the received optical power.

The BER performance after 10-km MCF is shown in Fig. 6. The received optical power is 7-dBm. With kernel mapping, the BER of all 7 cores can be reduced under the CI-BCH FEC limit, the KLMS performs similar as the Volterra filtering, while the KRLS gets better performance. The achieved line-rates for 7 cores are 80.51-Gbps, 72.86-Gbps, 73.36-Gbps, 87.91-Gbps, 78.24-Gbps, 71.77-Gbps, and 76.03-Gbps, respectively. The total system capacity with 10-km 7-core fiber is 540.68-Gbps (net-rate 505.31-Gbps).

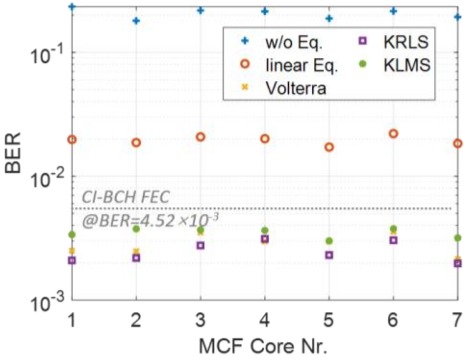

Fig. 6. BER after 10km 7-core fiber for different cores with 7-dBm received optical power.

The kernel mapping carries out infinite dimension calculation from HS to RKHS, which avoid the proposed KLMS and KRLS truncating operations. It is the main reason that the kernel mapping based methods could achieve similar or even better performance than the Volterra filtering scheme, which needs truncation operations in HS to guarantee the acceptable implementation complexity. When the modulation format changes from multi-carrier modulation to single-carrier modulation, such as the pulse amplitude modulation, the kernel mapping based equalization is still experimentally proved to be an effective method [25]. It is worth noticing that the kernel mapping method has a training process to optimize the parameters. The training process needs to be carried out when channel state changes, which do not frequently occur in static channels like the optical fiber channels.

Considering the storage complexity of kernel methods, the KLMS algorithm uses the stochastic gradient descent to obtain coefficients, and it has linear storage complexity in terms of the number of iterations, N_L, i.e. $\boldsymbol{O}(N_L)$ [17, 26]. The KRLS algorithm, on the other hand, calculates the coefficients by solving a least-square problem, involving the inversion of a kernel matrix, whose dimension depends on the number of iterations. Thus, it has quadratic storage complexity in terms of N_L, i.e. $\boldsymbol{O}(N_L^2)$ [17, 26]. Moreover, the number of multiplications of the kernel methods mainly depends on the size of the feature space F that is composed of feature vectors φ (.) of the RKHS. A factor that influences the size of F is the number of training overhead N_L. For the KLMS, its number of multiplications is equal to N_L, while the KRLS has the number of multiplications as N_L^2 [17, 27, 28]. Comparatively, the number of multiplications of the Volterra series equalizer is $\sum_{p=1}^{P}(\boldsymbol{M}_p)^p$, where M_p is the tap number (i.e. memory length) of the p-th order nonlinear kernels. Table. 1 shows the number of multiplications of different algorithms. According to the experimental results shown in Fig. 4, the BER performance becomes statured when the training overhead is more than 4096 for both the KLMS and KRLS. Therefore, we take this value to compare the kernel methods with the Volterra filtering.

Table. 1 Comparison of number of multiplications of Volterra filtering, KLMS and KRLS algorithm.

Algorithm	*Number of multiplications*	
	Formula	*The case for comparison*
Volterra	$\sum_{p=1}^{P}(\boldsymbol{M}_p)^p$	15595 (M_2=15, M_3=9, M_4=11)
KLMS	N_L	4096 (N_L=4096)
KRLS	N_L^2	16777216 (N_L=4096)

It can be seen that when the feature space increases the kernel functions requires a higher number of multiplication operations that directly reflect the computation complexity. Although the KRLS always requires less training overhead for a given level of the BER, its higher computation complexity may hinder the practical viability. In [17, 29], the authors have analyzed sparsification techniques, such as the novelty criterion (NC), the coherence criterion, the quantization criterion and surprise criterion, to reduce the size of feature space and thus decrease N_L. The theoretical analysis has shown sparsification techniques can greatly reduce the complexity of kernel mapping based equalization techniques. In [28], the number of multiplications of the KLMS has decreased from 5000 to 150 after NC sparsification. Therefore, sparsification techniques could be an interesting direction to develop kernel mapping for channel equalization in short-reach optical links. Moreover, the mapping process in Eq. (4a) uses exponentiation, which might require application specific integrated circuit (ASIC) design in the implementation. The authors in [30] exploited an exponentially large quantum state space through controllable entanglement and interference to use the quantum state space as the feature space. It is envisioned the exponentiation of the kernel mapping process could be well treated in the future quantum computers [30].

5. Conclusions

To conclude, we introduce ‘kernel trick’ to optical short-reach communication systems to efficiently mitigate nonlinear impairments. From the signal processing perspective, the kernel mapping uses inner production calculation instead of direct high-dimensional mapping scheme, addressing the nonlinear impairments in a high-dimensional manner while avoiding the ‘curse of dimension’ efficiently. For a short-reach IM/DD system, the experimental results have demonstrated that introducing kernel mapping achieves performance of the BER under the CI-BCH FEC limit that is comparable with the Volterra filtering scheme.

6. Funding and acknowledgments

6.1 Funding

Swedish Research Council (VR), the Swedish Foundation for Strategic Research (SSF), Göran Gustafsson Foundation, Swedish ICT-TNG, EU H2020 MCSA-IF Project NEWMAN (#752826), the Celtic-Plus sub-project SENDATE-EXTEND & SENDATE FICUS, H2020 5GPPP 5G-PHOS research program (ref.761989), European Commission through the FP7 project MIRAGE (ref.318228), SJTU State Key Laboratory of Advanced Optical Communication System and Networks Open Project 2018GZKF03001. Natural Science Foundation of China (#61331010, 61722108, 61775137, 61671212).

6.2 Acknowledgments

We thank J. Van Kerrebrouck, G. Torfs, J. Bauwelinck from Ghent University - IMEC, IDLab; G. Van Steenberge from the Centre for Microsystems Technology, Ghent University-imec; S. Spiga and M. C. Amann from the Walter Schottky Institut, Technische Universität München for the help of VCSEL design, fabrication and setup in the experiment. We also thank L. Gan, S. Fu and D. Liu from the Wuhan National lab for Optoelectronics, Huazhong University of Sci&Tech for the help of multi-core fiber fabrication and setup.

7. Appendix

7.1 Hilbert space (HS)

Considering an inner product signal space *H* with infinite orthonormal basis set $\{ \boldsymbol{x}_i^{\boldsymbol{b}} \}_{i=1\text{->}\infty}$, the signal vector $\boldsymbol{x}$ in *H* can be expressed as:

$$\boldsymbol{x} = \sum\nolimits_{i=1}^{\infty} a_i \boldsymbol{x}_i^{\boldsymbol{b}}, \tag{19}$$

where a_i is the coefficient. Once a_i meets the following conditions:

$$\sum\nolimits_{i=m+1}^{n} a_i^2 \to 0, \Big| \forall m,n \in R^+, m<n \,\&\, m,n \to +\infty, \tag{20a}$$

$$\sum\nolimits_{i=1}^{n} a_i^2 < \infty, \tag{20b}$$

the distance between the two vectors in the inner product space *H* becomes meaningful and the matrix norm is orthonormal. Such an inner product space *H* is complete and also referred to as HS. The conventional signal analysis and processing in optical communications are normally performed in the HS. The training data set at the *k*-th sampling point in time is assumed as $\{\boldsymbol{x}(k), d(k)\}|_{k=1,2,\ldots N}$, where $\boldsymbol{x}(k)$ is the received signal vector at the *k*-th sampling point in the HS and $d(k)$ is the transmitted signal at the *k*-th sampling point, *N* is the training data size. The target of the recovered transmitted signal over a fiber link at the receiver in the HS is:

$$\min \ \left\| d(k) - \boldsymbol{W}(k) \cdot \boldsymbol{x}(k) \right\|, \tag{21}$$

where $\boldsymbol{W}(k)$ is the linear transformation (i.e. coefficient matrix) of the received signal vector $\boldsymbol{x}(k)$.

7.2 Reproducing kernel Hilbert space (RKHS)

Different from the conventional equalization methods, when introducing the kernel mapping in optical short-reach communications the signal space is changed to the RKHS. We define a function $\kappa(\boldsymbol{x_i}, \boldsymbol{x_j})$, for which inputs are two *t*-dimension vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in *H*, and an output is a real number. $\kappa(\boldsymbol{x_i}, \boldsymbol{x_j})$is a kernel function if and only if the Gram matrix shown in (22a) is a positive semi-definitive matrix, which is defined as (22b).

$$\boldsymbol{K} = \begin{bmatrix} \kappa(\boldsymbol{x}_1, \boldsymbol{x}_1) & \dots & \kappa(\boldsymbol{x}_1, \boldsymbol{x}_m) \\ \dots & \dots & \dots \\ \kappa(\boldsymbol{x}_m, \boldsymbol{x}_1) & \dots & \kappa(\boldsymbol{x}_m, \boldsymbol{x}_m) \end{bmatrix}, \big| \forall m \in \boldsymbol{N}, \tag{22a}$$

$$\sum_{i=1}^{m} \sum_{j=1}^{m} a_i a_j \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0, \big| \forall m \in \boldsymbol{N}, \forall x_1 \dots \boldsymbol{x}_m \in \boldsymbol{H}, \forall a_1 \dots a_m \in \boldsymbol{R}. \tag{22b}$$

As a result, the kernel function κ is a continuous, symmetric and positive function defined in the space *H*, also referred to as ‘Mercer kernel’. Gaussian kernel [17] is one of representative kernel functions and can be expressed as:

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\left\| \boldsymbol{x}_i - \boldsymbol{x}_j \right\|^2 / 2\sigma^2). \tag{23}$$

The parameter σ is the bandwidth of the Gaussian kernel. Assume *H* to be any vector space of all real-valued functions generated by the kernel κ. Functions *f* and *g* from *H* are defined as follow:

$$f = \sum_{i=1}^{n} a_i \kappa(\boldsymbol{x}_i, \cdot), g = \sum_{j=1}^{m} \beta_j \kappa(\boldsymbol{x}_j, \cdot), \forall n, m \in \boldsymbol{N}, \tag{24}$$

where α_i and β_j are the coefficients of the vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ defined in the domain *H*. The bilinear form of *f* and *g* is defined as:

$$\left\langle f, g \right\rangle = \sum_{i=1}^{n} \sum_{j=1}^{m} a_i \beta_j \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j). \tag{25}$$

Since the bilinear form in (25) satisfies the symmetry, linearity and positive definitive properties [20], (25) can be considered as the inner product belonging to the space *H*. Besides, it also meets the condition of the ‘reproducing’ property, expressed as:

$$\left\langle f, \kappa(\cdot, \boldsymbol{x}_j) \right\rangle = \sum_{i=1}^{n} a_i \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = f(\boldsymbol{x}_j). \tag{26}$$

Thus, the kernel function defined in the space *H* meets the conditions of both the inner product and the ‘reproducing’ property. Such a function is recognized as the ‘reproducing kernel function’, and the space composed by the kernel functions is called RKHS. The vector signal $\boldsymbol{x}$ in the HS is mapped to the RKHS by the kernel function κ (. , $\boldsymbol{x}$), and the linear processing mechanism $f + g$ is transformed to an inner product calculation <*f* , *g*> . Instead of (21), in the RKHS the target of recovered signal over a fiber link at the *k*-th sampling point is:

$$\min \ \left\| d(k) - f(\boldsymbol{x}(k)) \right\|. \tag{27}$$

The optimal solution for solving (27) can be expressed as:

$$f = \sum_{k=1}^{N} a_k \kappa(\cdot, \boldsymbol{x}(k)), \tag{28}$$

where α_i is the optimized coefficients, and *N* is the length of training data $\{\boldsymbol{x}(k), d(k)\}|_{k=1,2,\dots N}$. Although the functions in the RKHS are expansions with an arbitrary number of variables, the result can be expressed only with the training data set [10, 12]. Hence, the optimization problem with an arbitrary number of variables is transformed into the one with *N* variables. Finally, we reach to (1) in Section II to continue the derivations.

References

1. Cisco global cloud index: forecast and methodology, 2015–2020.
2. 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,” J. Lightw. Technol., **36**(2), 377-400 (2018).
3. X. Pang, O. Ozolins, L. Zhang, A. Udalcovs, R. Lin, R. Schatz, U. Westergren, G. Jacobsen, S. Popov, and J. Chen, “Beyond 200 Gbps per Lane Intensity Modulation Direct Detection (IM/DD) Transmissions for Optical Interconnects: Challenges and Recent Developments,” in Proc. OFC (2019).

4. H. Yamazaki, M. Nagatani, H. Wakita, M. Nakamura, S. Kanazawa, M. Ida, T. Hashimoto, H. Nosaka, and Y. Miyamoto, "160-GBd (320-Gb/s) PAM4 Transmission Using 97-GHz Bandwidth Analog Multiplexer," IEEE Photon. Technol. Lett., **30**(20), 1749-1751 (2018).
5. H. Yamazaki, M. Nagatani, H. Wakita, Y. Ogiso, M. Nakamura, M. Ida, T. Hashimoto, H. Nosaka, and Y. Miyamoto, "Transmission of 400-Gbps Discrete Multi-tone Signal Using >100-GHz-Bandwidth Analog Multiplexer and InP Mach-Zehnder Modulator," in Proc. ECOC (2018).
6. S. Lange, S. Wolf, J. Lutz, L. Altenhain, R. Schmid, R. Kaiser, M. Schell, C. Koos, and S. Randel, "100 GBd Intensity Modulation and Direct Detection with an InP-Based Monolithic DFB Laser Mach–Zehnder Modulator," J. Lightw. Technol., **36**(1), 97-102 (2018).
7. N. Stojanovic C. Prodaniuc, L. Zhang, and J. Wei, "210/225 Gbit/s PAM-6 transmission with BER bellow KP4-FEC/EFEC and at least 14 dB link budget," in Proc. ECOC (2018).
8. L. Zhang, J. Wei, N. Stojanovic, C. Prodaniuc, and C. Xie, "Beyond 200-Gb/s DMT Transmission over 2-km SMF Based on A Low-cost Architecture with Single-wavelength, Single-DAC/ADC and Single-PD," in Proc. ECOC (2018).
9. X. Zhou, R. Urata, and H. Liu, "Beyond 1Tb/s datacenter interconnect technology: challenges and solutions," in Proc. OFC (2019).
10. J. Van Kerrebrouck, X. Pang, O. Ozolins, R. Lin, A. Udalcovs, L. Zhang, H. Li, S. Spiga, M. C. Amann, L. Gan, M. Tang, S. Fu, R. Schatz, G. Jacobsen, S. Popov, D. Liu, W. Tong, G. Torfs, J. Bauwelinck, J. Chen, and X. Yin, "High-speed PAM4-based optical SDM interconnects with directly modulated long-wavelength VCSEL," J. Lightw. Technol., **37**(2), 356-362, (2019).
11. G. P. Agrawal, Fiber-Optic Communication Systems, Hoboken, USA, Wiley 2002.
12. J. Zhang, J. Yu, J. Shi, and H. C. Chien, "Digital Dispersion Pre-compensation and Nonlinearity Impairment Pre- and Post-processing for C-band 400G PAM-4 Transmission over SSMF Based on Direct Detection," in Proc. ECOC (2017).
13. L. Zhang, X. Hong, X. Pang, O. Ozolins, A. Udalcovs, R. Schatz, C. Guo, J. Zhang, F. Nordwall, K. M. Engenhardt, U. Westergren, S. Popov, G. Jacobsen, S. Xiao, W. Hu, and J. Chen, "Nonlinearity-aware 200 Gbit/s DMT transmission for C-band short-reach optical interconnects with a single packaged electro-absorption modulated laser," Opt. Lett., **43**(2), 182-185 (2018).
14. L. Sun, J. Du, and Z. He, "Machine learning for nonlinearity mitigation in CAP modulated optical interconnect system by using K-nearest neighbour algorithm," in Proc. ACP (2016).
15. G. Chen, J. Du, L. Sun, W. Zhang, K. Xu, X. Chen, G. T. Reed, and Z. He, "Nonlinear Distortion Mitigation by Machine learning of SVM classification for PAM-4 and PAM-8 modulated optical interconnection," J. Lightw. Technol., **36**(3), 650-657 (2017).
16. E. Novak, and R. Klaus, "The curse of dimension and a universal method for numerical integration," Multivariate approximation and splines, 177-187 (1997).
17. W. Liu, C. P. Jose, and H. Simon, "Kernel adaptive filtering: a comprehensive introduction," 57, John Wiley & Sons (2011).
18. N. Aronszajn, "Theory of reproducing kernels," Trans. of the American Mathematical Society, **68**, 337 (1950).
19. L. Zhang, J. Van Kerrebrouck, O. Ozolins, R. Lin, X. Pang, A. Udalcovs, S. Spiga, MC Amann, G. Van Steenberge, L. Gan, M. Tang, S. Fu, R. Schatz, S. Popov, D. Liu, W. Tong, G. Torfs, J. Bauwelinck, X. Yin, S. Xiao, and J. Chen, "Experimental Demonstration of 503.61-Gbit/s DMT over 10-km 7-Core Fiber with 1.5-μm SM-VCSEL for Optical Interconnects," in Proc. ECOC (2018).
20. B. Schölkopf, R. Herbrich, A. Smola, and R. Williamson, "A generalized representer theorem," in Proc. of the Annual Conference on Computational Learning Theory, 416 (2001).
21. W. Klaus, B. J. Puttnam, R. S. Luís, J. Sakaguchi, J. M. Delgado Mendinueta, Y. Awaji, and N. Wada, "Advanced space division multiplexing technologies for optical networks," *J. Opt. Commun. Netw.*, **9**(4), C1-C11 (2017).
22. P. S. Chow, J. M. Cioffi, and J. A.C. Bingham, "A Practical Discrete Multitone Transceiver Loading Algorithm for Data Transmission over Spectrally Shaped Channels," IEEE Trans. Commun., **43**(234), 773-775 (1995).
23. M. Scholten, T. Coe, and J. Dillard, "Continuously-interleaved BCH (CIBCH) FEC delivers best in class NECG for 40G and 100G metro applications," in Proc. OFC (2010,).
24. E. Agrell, M. Secondini, "Information-Theoretic Tools for Optical Communications Engineers," in Proc. IPC (2018).
25. L. Zhang, O. Ozolins, R. Lin, A. Udalcovs, X. Pang, L. Gan, R. Schatz, A. Djupsjöbacka, J. Mårtensson, U. Westergren, M. Tang, S. Fu, D. Liu, W. Tong, S. Popov, G. Jacobsen, W. Hu, S. Xiao and J. Chen, "Kernel Adaptive Filtering for Nonlinearity-Tolerant Optical Direct Detection Systems," in Proc. ECOC (2018).
26. S. Van Vaerenbergh, and I. Santamaría, "A comparative study of kernel adaptive filtering algorithms," in Digital Signal Processing and Signal Processing Education Meeting, 181-186 (2013).
27. B. Chen, S. Zhao, P. Zhu, and J. C. Principe, "Quantized kernel recursive least squares algorithm," IEEE Transactions on Neural Networks and Learning Systems, **24**(9), 1484–1491 (2013).
28. U. Singh, R. Mitra, V. Bhatia, and A. Mishra, "Kernel LMS based Estimation Techniques for Radar Systems," IEEE Transactions on Aerospace and Electronic Systems, 2019.
29. J. Platt, "A resource - allocating network for function interpolation," Neural Computation, **3**(2), 213 – 225 (1991).

30. V. Havlíček, A. D. Córcoles, K. temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, “Supervised learning with quantum-enhanced feature spaces,” Nature, **567**(7747), 209-212 (2019).