

Equalization in chaos-based communication systems using kernel adaptive filtering

Renato Candido, Marcio Eisencraft, Magno T. M. Silva

Abstract—Chaos-based communication systems have attracted attention of researchers in academy and industry in the last decades. A particular family of such systems has as basic idea to use the transmitted message to modify a known nonlinear chaotic signal generator (CSG). In the receiver, the knowledge of the employed nonlinear CSG in conjunction with chaotic synchronization permits to recover the original message. These systems are an alternative for spread spectrum communication with a possible increase in the security in the physical layer, since it is necessary to perfectly know the CSG in the receiver to decode the message. However, the lack of robustness of chaotic synchronization in relation to channel noise and intersymbol interference still poses a barrier for their practical use. The problem of equalization for such systems have been tackled for a while, and algorithms based on the normalized least-mean squares have presented auspicious results for linear channels. For nonlinear channels, Kernel Adaptive Filters (KAFs) have been used since they are able to solve nonlinear problems implicitly projecting the input vector into a larger dimension space, where they can be linearly solved. Therefore, in this paper, we propose the use of KAFs with two purposes: to equalize linear and nonlinear channels and, at the same time, decode the message without knowledge of the CSG in the receiver. Simulation results show that the proposed solution is able to perform these tasks.

Index Terms—Chaos-based communication, adaptive signal processing, equalizers, kernel adaptive filtering.

I. INTRODUCTION

A deterministic chaotic signal generator (CSG) produces trajectories in the state space that are aperiodic, bounded, and present sensitive dependence on initial conditions (SDIC) [2]. Chaos-based communication systems (CBCSs) using the synchronization of CSGs have been extensively investigated in the literature (see, e.g., [3]–[15] and their references). Even some patents were issued regarding the subject, see e.g. [16]–[19]. Some works with a practical approach using chaotic synchronization have also appeared, mainly in the optical communication domain (see, e.g., [7], [8]). This is somewhat natural since chaotic generators can be easily created using the intrinsic nonlinear properties of lasers [8].

A particular family of CBCS has as basic idea to use the transmitted message to modify a known nonlinear CSG. In the receiver, the knowledge of the employed CSG in conjunction with chaotic synchronization permits to recover the original message. This way, chaotic signals can be used as broadband

carriers for information signals with the potential to introduce a level of secrecy in the physical layer, represented by the CSG's parameters [6], [7]. Some papers also show that chaotic signals are optimal communication waveforms in some contexts [20]. Therefore, CBCSs present some interesting features from the telecommunications point of view.

One of the major drawbacks of CBCS is the poor robustness of the chaotic synchronization in relation to the noise level and the intersymbol interference (ISI) introduced by the channel. Even a small noise level or simple distortions may be sufficient to hinder communication [11], [12], [14], [21], [22]. In Fig. 1, we show some results to illustrate the effects of a nonideal communication channel. We have considered the encoding of a binary message $m(n) \in \{-1; 1\}$ showed in Fig. 1-(a), leading to the transmission of the chaotic signal $s(n)$, showed in Fig. 1-(b), considering the CBCS used in [14], described in Sec. II. Considering an ideal communication channel, the message is recovered without any errors as showed in Fig. 1-(c) and Fig. 1-(d), despite the small distortion that occurs during the synchronization of the CSGs, as shown in the first samples of the decoded signal $y(n)$. However, when we consider a simple nonideal channel which attenuates the transmitted signal by the factor of 0.9, the receiver is not able to synchronize and decode the transmitted message, which causes errors as shown in Fig. 1-(e) and Fig. 1-(f).

In order to mitigate the ISI, equalization schemes applied to CBCSs have been proposed in the literature, using different approaches of message encoding (see, e.g., [10], [11], [14], [23]–[27]). Among these references, [10], [11] and [14] consider the equalization applied in the discrete-time domain for the chaotic modulation that feeds back the transmitted sequence in the CSG. However, the recovery of the message proposed in these references depends on the good performance of the equalizer, which in turn uses the chaotic map since the same CSG of the transmitter is used in the receiver.

As an alternative to the adaptive equalizers of [10], [11] and [14], it is proposed in this paper to use kernel adaptive filters (KAFs) in the receiver. KAFs are able to tackle nonlinear problems, implicitly projecting the input vector into a larger dimension space, where they can be linearly solved [28]. Thus, when using a KAF in a CBCS, it is possible to equalize nonlinear communication channels. Besides that, it is no longer necessary to know the CSG in the receiver since a KAF can synchronize and decode the transmitted message implicitly. Despite the advantage of not needing to adjust synchronization parameters in the receiver, this result may imply a challenge for the secrecy in CBCSs argument.

Preliminary parts of this work were published as conference

The work of Eisencraft was partly supported by CNPq under grant 309275/2016-4. The work of Silva was supported by FAPESP under grant 2017/20378-9 and by CNPq under grant 304715/2017-4.

The authors are with Escola Politécnica, Universidade de São Paulo, São Paulo, SP, Brazil; e-mails: renatocan@lps.usp.br, marcioft@usp.br, magno.silva@usp.br. ph. +55-11-3091-5134.

Digital Object Identifier: 10.14209/jcis.2019.6

paper [1] (in Portuguese). In this paper, we extend these results, using an adaptive algorithm to adjust the kernel width parameter. We also provide a more detailed comparison with previous approaches by means of new simulations, in which we consider bit error rate (BER) and squared error level as performance measurements.

The paper is organized as follows. The problem is formulated in Section II, where the CBCS and the KAF under consideration are described. Simulation results are shown in Section III. Section IV closes the paper with some conclusions and perspectives of future works.

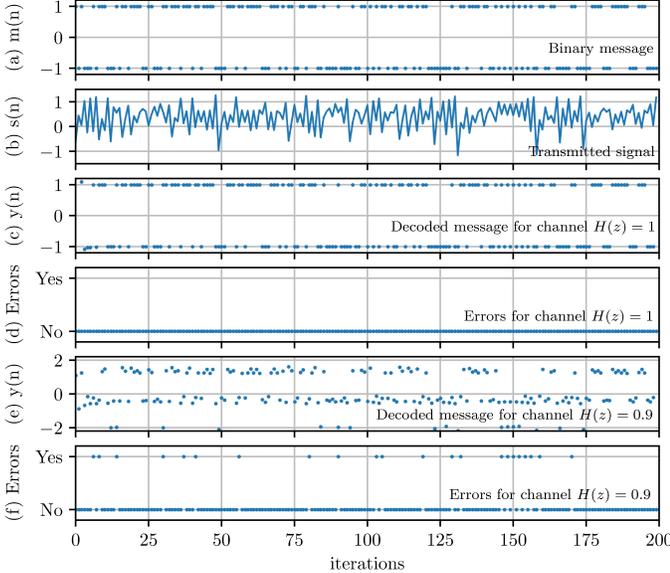


Fig. 1. CBCS and the effects of a nonideal communication channel. (a) binary message, (b) transmitted signal, (c) decoded message on an ideal channel, (d) errors after decision on an ideal channel, (e) decoded message on the channel with transfer function $H(z) = 0.9$, (f) errors after decision on the channel with transfer function $H(z) = 0.9$.

II. PROBLEM FORMULATION

In this section, we describe the CBCS considering a CSG in the receiver as [11], [14], [29]. In the sequel, we propose a new receiver based on a KAF that uses no information on the CSG. Finally, KAFs are described with focus on the quantized kernel least-mean squares algorithm.

A. The CBCS with a CSG in the receiver

Fig. 2 shows the CBCS considered in [11], [14], and [29], which is a discrete-time lowpass equivalent version of the one proposed in [30]. In this scheme, a binary sequence $m(n) \in \{-1, +1\}$ is encoded by using the first component of the master state vector $\mathbf{x}(n)$, via an encoding function

$$s(n) = c(x_1(n), m(n)). \quad (1)$$

Then, the signal $s(n)$ is fed back into the chaotic signal generator (CSG) and transmitted through a communication channel, whose model is constituted by a transfer function $H(z)$ and additive white Gaussian noise (AWGN). The receiver is constituted by an equalizer, a decoding function, and

the same CSG of the transmitter [11], [14], [29]. The equalizer should mitigate the ISI introduced by the channel in order to obtain the estimate of the transmitted signal, denoted as $\hat{s}(n)$, with an unavoidable delay of Δ samples. The decoding function is the inverse of (1) with respect to $m(n)$, i.e.,

$$y(n) = c^{-1}(\hat{x}_1(n), \hat{s}(n)), \quad (2)$$

where $\hat{x}_1(n)$ is the estimate of the state $x_1(n)$ and $y(n)$ is the estimate of the transmitted message $m(n - \Delta)$. Finally, $y(n)$ is fed back into the CSG of the receiver to obtain the estimate $\hat{x}_1(n)$.

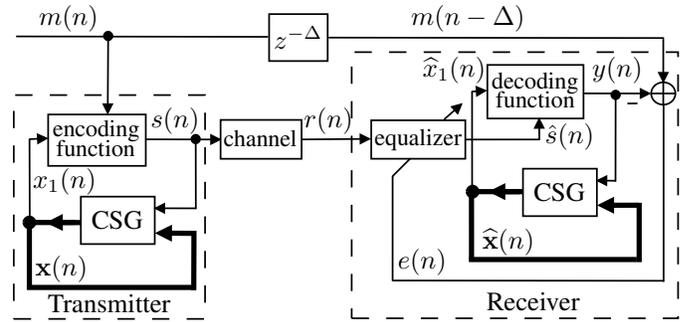


Fig. 2. Chaotic synchronization system with an adaptive equalizer.

As in [11], [14], the Hénon map [31] is used as CSG. Its equation is given by

$$\mathbf{x}(n+1) = \mathbf{A}\mathbf{x}(n) + \mathbf{b} + \mathbf{f}(s(n)), \quad (3)$$

where $\mathbf{x}(n) \triangleq [x_1(n) \ x_2(n)]^T$, $(\cdot)^T$ indicates transposition,

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ \beta & 0 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \mathbf{f}(s(n)) = \begin{bmatrix} -\alpha s^2(n) \\ 0 \end{bmatrix}, \quad (4)$$

being α and β real constant parameters of the map set as $\alpha = 1.4$ and $\beta = 0.3$ [2].

We also use the following encoding function

$$s(n) = \eta_1 x_1(n) - \eta_2 [m(n) + 1] \text{sign}[\eta_1 x_1(n)], \quad (5)$$

where $\text{sign}[\cdot]$ is the sign function. This function was proposed in [14] to reduce the disturbance caused by the message in $x_1(n)$ and increase the space of the parameters $\{\eta_1, \eta_2\}$, where the generated signal is chaotic. From (5), we can observe that, if $m(n) = -1$, $s(n) = \eta_1 x_1(n)$ and if $m(n) = 1$ a constant with the sign opposite to $\eta_1 x_1(n)$ is added to it. Fig. 3 shows the maximum Lyapunov exponent λ of the transmitted signal for this encoding function as a function of η_1 and η_2 considering a random equiprobable binary $m(n) \in \{-1, 1\}$. This Lyapunov exponent was obtained using the method described in [2, Sec. 5.2], considering $m(n)$ as a variable parameter. The region where λ is negative, i.e., the generated signals are not chaotic, is shown in gray. The colored region indicates the area where the generated signals are chaotic ($\lambda > 0$, i.e., the signals present SDIC). For the values of η_1 and η_2 in the white area of the figure, the transmitter diverges. To ensure the transmission of chaotic signals and still obtain a good equalization performance, we consider $\eta_1 = 0.9$ and $\eta_2 = 0.3$ (point indicated in the figure) in all simulations of this paper, as done in [14].

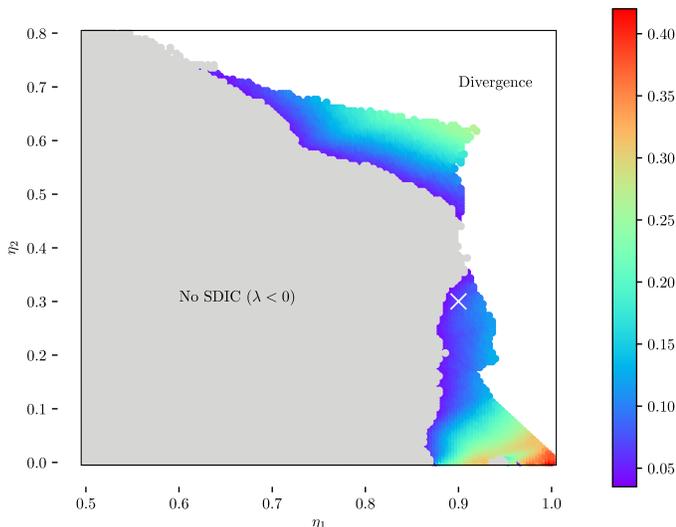


Fig. 3. Maximum Lyapunov exponent λ obtained when using (5) as a function of η_1 and η_2 . The point indicated by the \times represent the values of η_1 and η_2 considered in the simulations, to ensure the transmission of chaotic signals and still obtain a good equalization performance.

In order to update the coefficients of the equalizer, a normalized version of the least-mean squares (LMS) algorithm was proposed in [11], by encoding the transmitted sequence as

$$s(n) = x_1(n)m(n). \quad (6)$$

This algorithm, denoted as cNLMS $_{\times}$ (chaotic normalized LMS), minimizes the instantaneous cost function $\hat{J}(n) = e^2(n)$ using the stochastic gradient descent rule, where $e(n)$ is the estimation error, defined as

$$e(n) = m(n - \Delta) - y(n). \quad (7)$$

The cNLMS $_{\times}$ algorithm was extended in [14] to take into account the encoding function (5) and the resulting algorithm was denoted as cNLMS $_{+}$. As observed in [14], the encoding function (5) and the cNLMS $_{+}$ algorithm provides better performance in terms of BER and mean-square error (MSE) than that of cNLMS $_{\times}$ with (6). Therefore, we consider (5) with cNLMS $_{+}$ in the comparisons of this paper.

B. The CBCS with a KAF in the receiver

In this paper, we use a KAF for performing a double task in the CBCS: (i) mitigate the ISI introduced by the communication channel and (ii) invert the chaos-based transmitter. Thus, the transmitted message can be recovered at the receiver with no knowledge of the CSG. Fig. 4 shows the CBCS, in which the receiver is constituted only by a KAF with input column-vector $\mathbf{r}(n) = [r(n) \ r(n-1) \ \dots \ r(n-M+1)]^T \in \mathbb{R}^M$ and output $y(n)$. We assume that the KAF works in the training mode, where the transmitted sequence $m(n - \Delta)$ is known *a priori* in the receiver. Thus, $m(n - \Delta)$ plays the role of the desired sequence $d(n)$ and the estimation error defined in (7) is used to update the filter. We should notice that the KAF can be switched to the decision-directed mode, by replacing $m(n - \Delta)$ by the output of a decision device [32], [33].

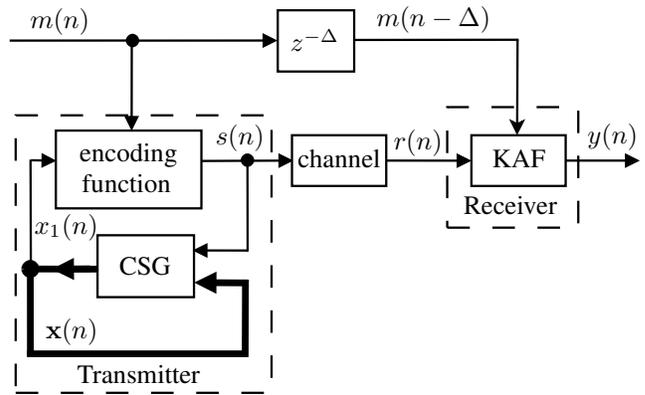


Fig. 4. Chaos-based communication system with a KAF as receiver.

C. Kernel adaptive filtering

Kernel adaptive filters have been mainly applied in system identification [34], echo cancellation [35], time series prediction [28], [36]–[38], and channel equalization [28], [39]. Although KAFs may present a superior performance when compared to the non-kernelized counterparts for solving nonlinear problems, their main drawbacks are the selection of an appropriate kernel and the high computational burden and memory used, since the dictionary size grows linearly with the incoming samples.

The LMS algorithm is the most popular adaptive filter due to its simplicity and robustness [40]. Its kernelized version, denoted as KLMS [39], is also the most popular among the different kernel adaptive filters in the literature. KLMS maps the input column-vector $\mathbf{r}(n) \in \mathbb{R}^M$ into a high dimensional feature space \mathbb{F} as $\varphi(\mathbf{r}(n))$, using a Mercer's kernel. Mercer's kernel is a continuous, symmetric, and positive-definite function $\kappa : \mathbb{R}^M \times \mathbb{R}^M \rightarrow \mathbb{R}$, such that

$$\kappa(\mathbf{r}, \mathbf{r}') = \varphi(\mathbf{r})^T \varphi(\mathbf{r}') \quad (8)$$

holds [28]. Eq. (8) is known as kernel trick since the inner product of the transformed feature vectors $\varphi(\mathbf{r})$ and $\varphi(\mathbf{r}')$ can be computed efficiently in the feature space \mathbb{F} through the kernel evaluations [28]. Consequently, the LMS algorithm can be used in the space \mathbb{F} to update the filter weight column-vector $\mathbf{\Omega}(n-1)$ in order to estimate the desired signal $d(n) \in \mathbb{R}$, that depends nonlinearly on the input $\mathbf{r}(n)$, i.e.,

$$\mathbf{\Omega}(0) = \mathbf{0}, \quad (9)$$

$$e(n) = d(n) - \varphi(\mathbf{r}(n))^T \mathbf{\Omega}(n-1), \quad (10)$$

$$\mathbf{\Omega}(n) = \mathbf{\Omega}(n-1) + \mu e(n) \varphi(\mathbf{r}(n)), \quad (11)$$

where μ is a step size. Fig. 5 shows a scheme for this algorithm considering equalization of communication channels.

From simple algebraic manipulations, we can write

$$\mathbf{\Omega}(n) = \sum_{i=1}^n \mu e(i) \varphi(\mathbf{r}(i)), \quad (12)$$

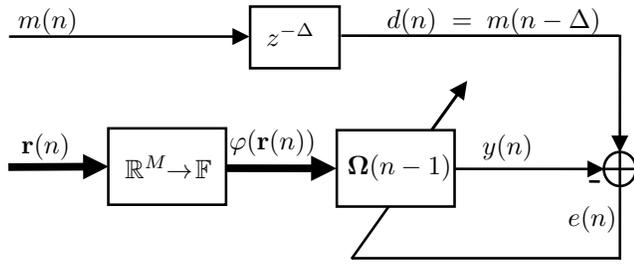


Fig. 5. KLMS applied to supervised equalization: $\mathbf{r}(n)$ is the input regressor vector constituted by samples of the output of the channel and $m(n)$ is the transmitted sequence.

which allows us to rewrite the estimate for $d(n)$ as

$$y(n) = \varphi(\mathbf{r}(n))^T \boldsymbol{\Omega}(n-1) = \sum_{i=1}^{n-1} \mu e(i) \kappa(\mathbf{r}(n), \mathbf{r}(i)). \quad (13)$$

Using (13) and defining $a(i) \triangleq \mu e(i)$, the KLMS algorithm can be summarized as

$$a(1) = \mu d(1), \quad (14)$$

$$y(n) = \sum_{i=1}^{n-1} a(i) \kappa(\mathbf{r}(n), \mathbf{r}(i)), \quad (15)$$

$$e(n) = d(n) - y(n), \quad (16)$$

$$a(n) = \mu e(n). \quad (17)$$

Throughout this paper, we focus on the Gaussian kernel, which is the most commonly used kernel in the literature and is defined as

$$\kappa(\mathbf{r}, \mathbf{r}') = \exp(-\zeta \|\mathbf{r} - \mathbf{r}'\|^2), \quad (18)$$

where $\zeta = 1/(2\sigma^2) > 0$ is the kernel parameter and σ is the kernel width [28].

One of the main challenges of kernel adaptive filtering is to reduce the dictionary size in order to minimize the computational complexity. The dictionary, also known as codebook, is defined as the set of support vectors $\mathcal{C}(n-1) = \{\mathbf{r}(c_j)\}_{j=1}^{N_c(n-1)}$, where $\mathbf{r}(c_j)$ is the j^{th} element and $N_c(n-1)$ is its cardinality, which can vary with n . The index $c_j \in \{1, 2, \dots, n-1\}$ is used to distinguish the dictionary elements $\mathbf{r}(c_1), \dots, \mathbf{r}(c_{N_c(n-1)})$ from the input vector $\mathbf{r}(n)$. With no dictionary restriction, we can observe from (13) that to estimate $d(n)$, we should use all the input vectors since the algorithm initialization until the previous time instant, i.e., $\mathcal{C}(n-1) = \{\mathbf{r}(i)\}_{i=1}^{n-1}$ and, in this case, the dictionary size $N_c(n-1) = n-1$ grows linearly with the incoming samples. To avoid this linear growth, different sparsification techniques were proposed in the literature to update the dictionary by including in it only informative data as are the cases of the novelty criterion [41] and the coherence criterion [42].

An efficient alternative to these techniques is the quantized KLMS (QKLMS) algorithm, which was obtained in [43] by quantizing the feature vector $\varphi(\mathbf{r}(n))$ in the update equation (11). To decide if the input vector must or must not be included

in the dictionary, we define

$$\text{dis}(\mathbf{r}(n), \mathcal{C}(n-1)) = \min_{1 \leq j \leq N_c(n-1)} \|\mathbf{r}(n) - \mathbf{r}(c_j)\|,$$

where $\|\cdot\|$ denotes the Euclidean norm. If $\text{dis}(\mathbf{r}(n), \mathcal{C}(n-1)) \leq \varepsilon$, being ε a threshold, the dictionary is kept unchanged and the coefficient of the closest vector is updated as

$$a(j^*) \leftarrow a(j^*) + \mu e(n),$$

where

$$j^* = \arg \min_{1 \leq j \leq N_c(n-1)} \|\mathbf{r}(n) - \mathbf{r}(c_j)\|.$$

Otherwise, the input vector is included in the dictionary and its respective coefficient stored in the memory, i.e., $\mathcal{C}(n) = \{\mathcal{C}(n-1), \mathbf{r}(n)\}$ and $a(N_c(n-1)+1) = \mu e(n)$. This procedure leads to an algorithm similar to the sparsified KLMS with novelty criterion. The main difference is that QKLMS uses the ‘‘redundant’’ data to locally update the coefficient of the closest vector. This coefficient update can enhance the use efficiency of that vector. Therefore, better accuracy and more compact network can be achieved [43].

To select an appropriate kernel, some adaptive multiple kernel approaches have been proposed in the literature (see, e.g., [37], [38], [44]–[48]). The main drawback of these solutions are the computational cost since many of them consider two or more kernel filters in parallel, each one with a different kernel function. Recently, some adaptive solutions to find a proper width for the Gaussian kernel were proposed in the literature [49], [50]. These solutions seek the minimum of the instantaneous squared error $\hat{J}(n) = e^2(n)$, using the stochastic gradient descent rule for updating the parameter ζ . Particularly, the exponent gradient variable normalization (EGVN) algorithm [50] updates ζ in the space of real positive numbers through

$$\zeta(n) = \zeta(n-1) \exp\left(-\mu_\zeta \zeta(n-1) \frac{\partial \hat{J}(n)}{\partial \zeta(n-1)}\right) \quad (19)$$

where μ_ζ is a step size and

$$\frac{\partial \hat{J}(n)}{\partial \zeta(n-1)} = 2e(n) \sum_{j=1}^{N_c(n)} \left\{ a_j(n-1) \|\mathbf{r}(n) - \mathbf{r}(c_j)\|^2 \times \exp(-\zeta(n-1) \|\mathbf{r}(n) - \mathbf{r}(c_j)\|^2) \right\}. \quad (20)$$

This algorithm, proposed in [50], improves the exponential gradient updates, since it is numerically stable even when ζ is small and avoids a slow adaptation when ζ is large.

The summary of the QKLMS algorithm with EGVN for adaptation of ζ is shown in Table I, where $a(j)$ is the j^{th} element of the coefficient vector \mathbf{a} . Due to its inherent advantages, the algorithm of this table is used in all simulations of this paper.

III. SIMULATION RESULTS

In order to verify the behavior of the proposed system, we have performed some numerical simulations. At first, we verify if the QKLMS algorithm is able to decode a binary message $m(n) \in \{-1; +1\}$ encoded by the chaos-based transmitter.

TABLE I
SUMMARY OF THE QKLMS ALGORITHM WITH EGVN FOR UPDATING
THE GAUSSIAN KERNEL PARAMETER.

<p>Input: $\{\mathbf{r}(n) \in \mathbb{R}^M, d(n)\}$, $n = 1, 2, \dots$ Initialization: Choose step sizes $\mu > 0$ and $\mu_\zeta \geq 0$, kernel parameter $\zeta(1) > 0$, threshold $\varepsilon \geq 0$, and initialize the dictionary $\mathcal{C}(1) = \{\mathbf{r}(1)\}$, $N_c(1) = 1$, and coefficient vector $\mathbf{a}(1) = [\mu d(1)]$.</p>
<p>For $n = 2, 3, \dots$, do:</p> <ol style="list-style-type: none"> 1) Compute the output of the adaptive filter: $y(n) = \sum_{j=1}^{N_c(n-1)} a_j(n-1) \kappa(\mathbf{r}(n), \mathbf{r}(c_j)),$ where $\kappa(\mathbf{r}(n), \mathbf{r}(c_j)) = \exp(-\zeta(n-1) \ \mathbf{r}(n) - \mathbf{r}(c_j)\ ^2)$ 2) Compute the estimation error: $e(n) = d(n) - y(n)$ 3) Compute the gradient for updating ζ: $\frac{\partial \hat{\mathcal{J}}(n)}{\partial \zeta(n-1)} = 2e(n) \sum_{j=1}^{N_c(n-1)} \left\{ a_j(n-1) \ \mathbf{r}(n) - \mathbf{r}(c_j)\ ^2 \times \kappa(\mathbf{r}(n), \mathbf{r}(c_j)) \right\}$ 4) Update ζ: $\zeta(n) = \zeta(n-1) \exp\left(-\mu_\zeta \zeta(n-1) \frac{\partial \hat{\mathcal{J}}(n)}{\partial \zeta(n-1)}\right)$ 5) Compute the distance between $\mathbf{r}(n)$ and $\mathcal{C}(n-1)$: $\text{dis}(\mathbf{r}(n), \mathcal{C}(n-1)) = \min_{1 \leq j \leq N_c(n-1)} \ \mathbf{r}(n) - \mathbf{r}(c_j)\$ 6) Decide if $\mathbf{r}(n)$ will be included in the dictionary: If $\text{dis}(\mathbf{r}(n), \mathcal{C}(n-1)) \leq \varepsilon$, keep the dictionary unchanged, i.e., $\mathcal{C}(n) = \mathcal{C}(n-1)$, and quantize $\mathbf{r}(n)$ to the closest vector by updating the coefficient of that vector, i.e., $a(j^*) \leftarrow a(j^*) + \mu e(n),$ where $j^* = \arg \min_{1 \leq j \leq N_c(n-1)} \ \mathbf{r}(n) - \mathbf{r}(c_j)\$ Otherwise, assign a new vector and the corresponding new coefficient: $\mathcal{C}(n) = \{\mathcal{C}(n-1), \mathbf{r}(n)\}$ $\mathbf{a}(n) = [\mathbf{a}(n-1), \mu e(n)]$ $N_c(n) = N_c(n-1) + 1$ <p>end</p>

For that, we consider a CBCS as shown in Fig. 4 with an ideal communication channel in the absence of noise. The results for this scenario are shown in Fig. 6. In this case, the chaotic encoder can be interpreted as a nonlinear channel and the KAF acts as a chaos-based receiver. By equalizing this channel, the KAF decodes the transmitted message from $r(n)$, which is a chaotic signal, obtaining the output shown in Fig. 6-(a) with almost no errors, as shown in Fig. 6-(b). In Fig. 6-(c), the kernel parameter $\zeta(n)$, initialized as $\zeta(0) = 50$ ($\sigma(0) = 0.1$) is shown along the iterations. It is possible to notice that the adaptation of this parameter is important, since it converges to a different value, around 13 in this case. In Fig. 6-(d), the size of the dictionary $N_c(n)$ along the iterations is shown. For

this scenario, about 6000 elements are necessary to decode the transmitted message due to the severe nonlinearity introduced by the chaos-based transmitter. It is worth to notice that the receiver does not have any information about the parameters used in the transmitter such as the chaotic map or the encoding function. As a metric of the performance of the algorithm, we consider the mean squared error (MSE) given by $E\{e^2(n)\}$. In Fig. 6-(e), an estimate of the MSE is shown considering an ensemble average of 50 independent runs, reaching -30 dB at $n = 10^5$.

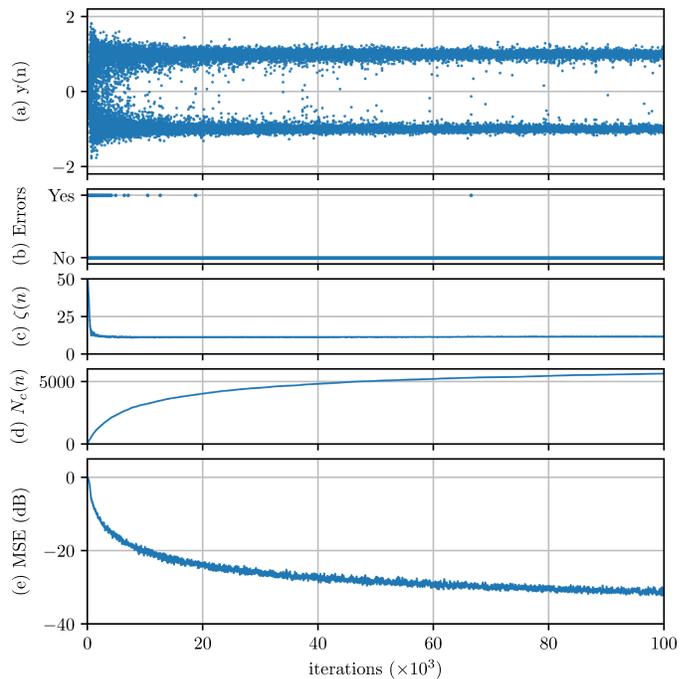


Fig. 6. CBCS using the QKLMS algorithm considering an ideal channel in the absence of noise ($M = 5$, $\varepsilon = 0.1$, $\mu = 0.5$, $\mu_\zeta = 0.01$, $\Delta = 1$). (a) Recovered sequence, (b) recovery errors, (c) evolution of the kernel parameter $\zeta(n)$, (d) size of the dictionary $N_c(n)$, and (e) estimated MSE considering an ensemble average of 50 independent runs. The MSE curve was filtered by a simple moving-average filter of 64 coefficients.

Next, we consider an equalization scenario with a CBCS system, to verify the performance of the QKLMS algorithm. A binary message $m(n) \in \{-1; +1\}$ is transmitted through a linear communication channel with transfer function

$$H_1(z) = 0.25 + z^{-1} + 0.25z^{-2} \quad (21)$$

from instant $n = 0$ until $n = 50 \times 10^3$. After that, the channel is abruptly changed to a nonlinear channel composed of a linear part with transfer function

$$H_2(z) = 0.1 + z^{-1} + 0.1z^{-2} \quad (22)$$

followed by a cubic nonlinear part [51], [52] in which the output is given applying the function

$$f(x) = x + 0.1x^2 + 0.1x^3 \quad (23)$$

to the input, considering a signal-to-noise ratio (SNR) of 40 dB. It is worth to notice that the linear part of this channel, with transfer function given by (22) introduces a lower level

of ISI than the linear channel (21), emphasizing the nonlinear effect, given by (23).

The results are shown in Figs. 7-(a) to 7-(e). In Figs. 7-(f) to 7-(j), as a performance benchmark, we show the results obtained with a conventional communication system without using chaos, but considering the same equalization scenario. In the conventional communication system, we can notice that the equalizer is able to mitigate the ISI introduced by the channels, recovering the transmitted message for both channels as shown by the output $y(n)$ in Fig. 7-(f) and the recovery errors shown in Fig. 7-(g). Again, the parameter $\zeta(n)$ was initialized with $\zeta(0) = 50$ ($\sigma(0) = 0.1$) and we observe that the value of the parameter $\zeta(n)$ changes along the iterations and has an abrupt variation when the communication channel is changed, which indicates that the optimum value of this parameter depends on the communication channel. In Fig. 7-(i), we observe that, for this scenario, about 120 elements are necessary to equalize the first channel and about 280 elements are necessary after the channel is changed. In the CBCS, the nonlinearity introduced by the chaos-based transmitter along with the distortion introduced by the communication channel represent a much more challenging situation for the equalizer. However, QKLMS is able to recover the transmitted sequence, by equalizing the channels and decoding the message as we can observe by the output shown in Fig. 7-(a) and the errors shown in Fig. 7-(b), reaching an MSE of about -20 dB at $n = 50 \times 10^3$. In this case, due to the distortion introduced by the chaotic-transmitter along with the communication channel, the size of the dictionary grows to about 9000 before the abrupt change of the communication channel and to about 16000 afterwards, as shown in Fig. 7-(d). This large number of elements in the dictionary causes a high increase in the computational cost of the algorithm, when compared to the solution presented in [11], [29].

We also compare the performance of the CBCS using the QKLMS algorithm with the performance of the cNLMS₊ from [14], considering the same scenario of Fig. 7. The results are shown in Fig. 8. It can be noticed that the cNLMS₊ algorithm presents a better performance between $n = 0$ and $n = 50 \times 10^3$, when a linear communication channel is considered. However, for $n > 50 \times 10^3$, when a nonlinear communication channel is considered, cNLMS₊ is not able to obtain a good result, since it considers a linear equalizer followed by a chaos-based decoder. On the other hand, QKLMS can equalize nonlinear communication channels, without any knowledge of the parameters of the chaos-based transmitter at the burden of a higher computational cost. We should mention that the delay Δ was set to achieve the best performance of each algorithm. It is well known that this parameter is essential to achieve good equalization results and can be different for each scheme due to their different manners to solve the problem.

In Table II, we show a performance comparison in terms of squared error level and BER between the CBCS with QKLMS, the CBCS with cNLMS₊, and the conventional communication system with QKLMS, considering the linear channel (21) and the nonlinear channel composed by (22) and (23). For each line of the table, the respective algorithm has been run for 300×10^3 iterations for convergence. Then,

the obtained parameters (the dictionary for QKLMS or the equalizer coefficients for the cNLMS₊) were kept fixed and 10^6 iterations were run to obtain the squared error level and the BER. The parameters used during convergence were the same of Figs. 7 and 8. In some cases, during the 10^6 iterations used to obtain the BER, no bit errors were observed, so that we could infer that the BER was lower than 10^{-6} .

TABLE II
PERFORMANCE COMPARISON OF A CBCS WITH QKLMS, A CBCS WITH cNLMS₊, AND A CONVENTIONAL COMMUNICATION SYSTEM WITH QKLMS ON THE LINEAR CHANNEL (21) AND THE NONLINEAR CHANNEL COMPOSED BY (22) AND (23) WITH AN SNR OF 40 DB.

System	Squared Error Level	BER
CBCS, QKLMS, Linear Channel	-32 dB	$10^{-5.1}$
CBCS, cNLMS ₊ , Linear Channel	-23 dB	$10^{-4.6}$
Conv., QKLMS, Linear Channel	-43 dB	$< 10^{-6}$
CBCS, QKLMS, Nonlinear Channel	-36 dB	$< 10^{-6}$
CBCS, cNLMS ₊ , Nonlinear Channel	-13 dB	10^{-2}
Conv., QKLMS, Nonlinear Channel	-37 dB	$< 10^{-6}$

When comparing the squared error levels shown in Table II with the respective MSE levels obtained in Figs. 7 and 8, we observe the values shown in the table are lower than the observed MSEs in the figures. This occurs since the results shown in the table were obtained using more iterations for convergence than the results shown in the figures. (300×10^3 instead of 50×10^3). Besides that, Figs. 7 and 8 consider an ensemble average of 50 runs to estimate the MSE whereas the results shown in Table II consider a single run to estimate the squared error level. During each of the 50 runs considered, the algorithms may perform poorly at different moments, causing the increase of the estimated MSE.

For the linear channel, we can observe that there were no significant performance difference between the CBCS with QKLMS, for which an MSE of -32 dB and a BER of $10^{-5.1}$ were obtained and the CBCS with cNLMS₊, for which an MSE of -23 dB and a BER of $10^{-4.6}$ were obtained. As expected, the conventional communication system provided the best performance with an MSE of -43 dB and no bit errors in 10^6 iterations, since it represents the least severe situation to equalize. For the nonlinear channel, we can observe that the performance of the CBCS with QKLMS, with an MSE of -36 dB and no bit errors in 10^6 iterations, was superior to the performance of the CBCS with cNLMS₊, which presented an MSE of -13 dB and a BER of 10^{-2} . This difference is due to the fact that cNLMS₊ is not capable of equalizing an arbitrary nonlinear channel, since it was designed to equalize a channel represented by the chaos-based encoding followed by a linear channel. Furthermore, different from the linear-channel case, a surprising result is observed: the CBCS with QKLMS performs close to the benchmark conventional communication system, which presented an MSE of -37 dB and no bit errors in 10^6 iterations.

To illustrate the importance of the correct adjustment of the parameter $\zeta(n)$, we present in Figs. 9 and 10 the results

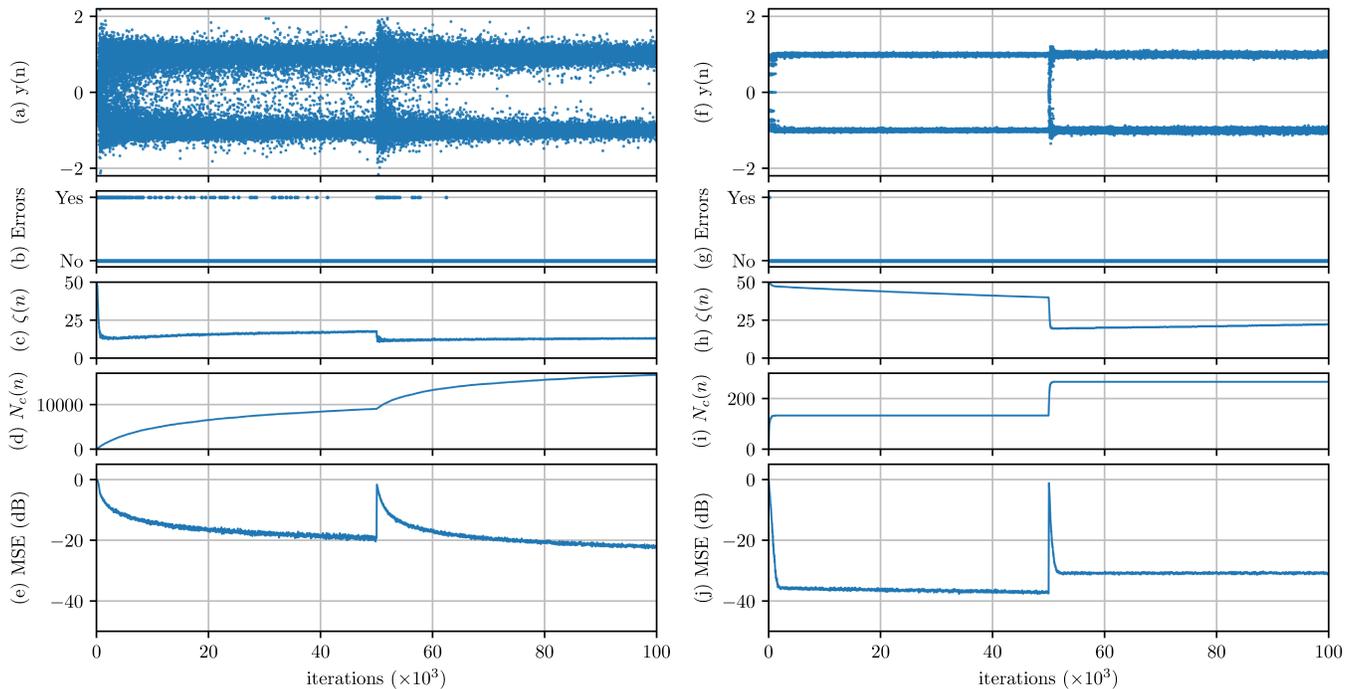


Fig. 7. CBCS and conventional communication system without using chaos using the QKLMS algorithm considering the linear channel (21) from $n = 0$ until $n = 50 \times 10^3$ and the nonlinear channel composed by (22) and (23) for $n > 50 \times 10^3$ with an SNR of 40 dB. CBCS: (a) Recovered sequence, (b) recovery errors, (c) $\zeta(n)$, (d) $N_c(n)$ obtained with QKLMS algorithm ($M = 5$, $\varepsilon = 0.1$, $\mu = 0.5$, $\mu_\zeta = 0.01$, $\Delta = 1$), and (e) estimated MSE considering an ensemble average of 50 independent runs. Conventional communication system without using chaos: (f) Recovered sequence, (g) recovery errors, (h) $\zeta(n)$, (i) $N_c(n)$ obtained with QKLMS algorithm ($M = 5$, $\varepsilon = 0.1$, $\mu = 0.5$, $\mu_\zeta = 0.01$, $\Delta = 1$), and (j) estimated MSE considering an ensemble average of 50 independent runs. The MSE curves were filtered by a simple moving-average filter of 64 coefficients.

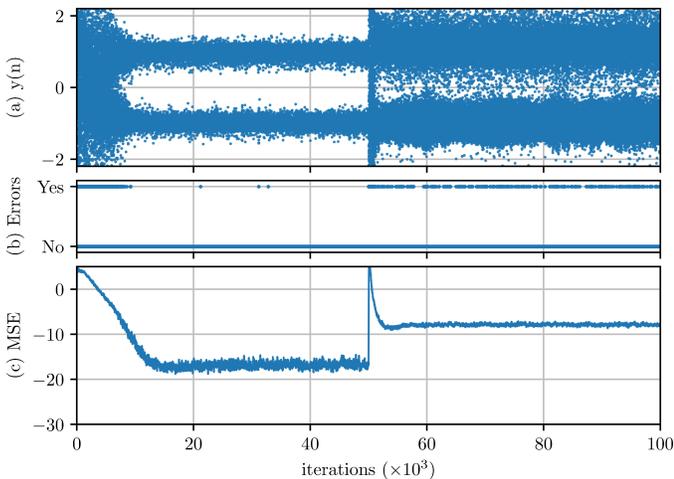


Fig. 8. CBCS using the cNLMS₊ algorithm considering the linear channel (21) from $n = 0$ until $n = 50 \times 10^3$ and the nonlinear channel composed by (22) and (23) for $n > 50 \times 10^3$ with an SNR of 40 dB. (a) Recovered sequence, (b) recovery errors obtained with cNLMS₊ from [14] ($M = 5$, $\varepsilon = 0.1$, $\tilde{\mu} = 0.01$, $\mu_\zeta = 0.01$, $\delta = 10^{-5}$, $\Delta = 3$), and (c) Estimated MSE considering an ensemble average of 50 independent runs. The MSE curve was filtered by a simple moving-average filter of 64 coefficients.

obtained with the CBCS in the same scenario used in Fig. 7 but considering a constant $\zeta(n)$, without adaptation, equals 0.5 and 0.005 respectively. As it can be noticed, both configurations lead to poor results with a high number of errors and a high

MSE level. These results emphasize the importance of using an adaptive algorithm to adjust the parameter $\zeta(n)$, such as EGVN.

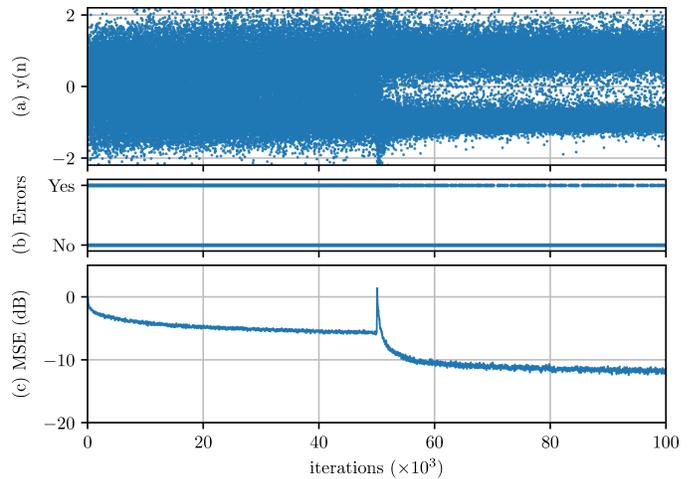


Fig. 9. CBCS using the QKLMS algorithm considering the linear channel (21) from $n = 0$ until $n = 50 \times 10^3$ and the nonlinear channel composed by (22) and (23) for $n > 50 \times 10^3$ with an SNR of 40 dB. (a) Recovered sequence, (b) recovery errors obtained with QKLMS algorithm considering $\zeta(n) = 0.5$ ($M = 5$, $\varepsilon = 0.1$, $\mu = 0.5$, $\Delta = 1$), and (c) Estimated MSE considering an ensemble average of 50 independent runs. The MSE curve was filtered by a simple moving-average filter of 64 coefficients.

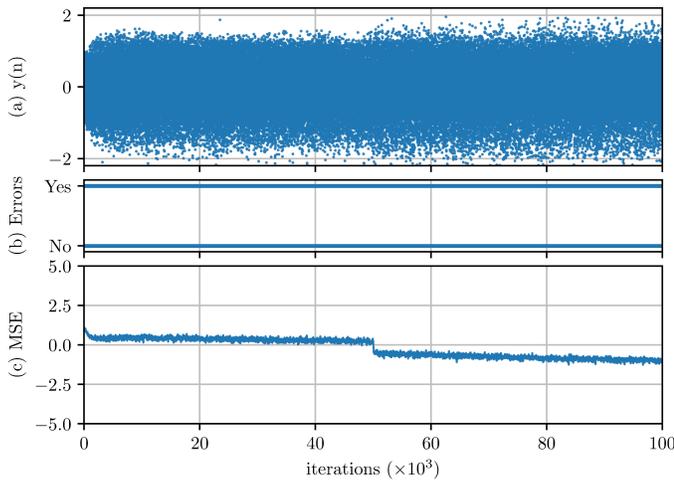


Fig. 10. CBCS using the QKLMS algorithm considering the linear channel (21) from $n = 0$ until $n = 50 \times 10^3$ and the nonlinear channel composed by (22) and (23) for $n > 50 \times 10^3$ with an SNR of 40 dB. (a) Recovered sequence, (b) recovery errors obtained with QKLMS algorithm considering $\zeta(n) = 0.005$ ($M = 5$, $\varepsilon = 0.1$, $\mu = 0.5$, $\Delta = 1$), and (c) Estimated MSE considering an ensemble average of 50 independent runs. The MSE curve was filtered by a simple moving-average filter of 64 coefficients.

IV. CONCLUSIONS

In this work, the message transmitted by a CBCS was recovered at the receiver by using a supervised adaptive kernel filter. Simulation results show that this filter is able to recover the transmitted message with no knowledge of the chaotic map and the coding function at the expense of a higher computational cost. For an ideal channel, the kernel filter can also be used to decode the message, playing the role of the slave in the CBCS. This may imply a challenge for the secrecy issue in CBCSs argument, which will be explored in a future work. Another extension we intend to investigate is the possibility of avoiding an overgrowth of the computational cost of KAF by removing elements of the dictionary [50] and of improving the obtained results by considering deep neural networks.

REFERENCES

- [1] R. Candido, M. T. M. Silva, and M. Eisencraft, "Equalização em sistemas de comunicação que utilizam sinais caóticos com algoritmos adaptativos baseados em kernel," in *Proc. of 7th Signal Processing Symposium (SPS)*, (<http://eventos.ufabc.edu.br/siimsp/anaais.html>), São Bernardo do Campo, Brazil, UFABC, 2017.
- [2] K. T. Alligood, T. Sauer, and J. A. Yorke, *Chaos: An Introduction to Dynamical Systems*. Textbooks in Mathematical Sciences, New York: Springer-Verlag, 1997.
- [3] L. M. Pecora and T. L. Carroll, "Synchronization in chaotic systems," *Phys. Rev. Lett.*, vol. 64, pp. 821–824, Feb. 1990.
- [4] A. V. Oppenheim, G. W. Wornell, S. H. Isabelle, and K. M. Cuomo, "Signal processing in the context of chaotic signals," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Process.*, vol. 4, pp. 117–120, 1992, doi: 10.1109/ICASSP.1992.226472.
- [5] K. M. Cuomo and A. V. Oppenheim, "Chaotic signals and systems for communications," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Process.*, vol. 3, pp. 137–140, Apr. 1993, doi: 10.1109/ICASSP.1993.319454.
- [6] M. Feki, B. Robert, G. Gelle, and M. Colas, "Secure digital communication using discrete-time chaos synchronization," *Chaos, Solitons & Fractals*, vol. 18, pp. 881–890, Nov. 2003, doi: 10.1016/S0960-0779(03)00065-1.

- [7] A. Argyris, D. Syvridis, L. Larger, V. Annovazzi-Lodi, P. Colet, I. Fischer, J. Garcia-Ojalvo, C. R. Mirasso, L. Pesquera, and K. A. Shore, "Chaos-based communications at high bit rates using commercial fibre-optic links," *Nature*, vol. 438, no. 7066, pp. 343–346, 2005, doi: 10.1038/nature04275.
- [8] A. Uchida, *Optical Communication with Chaotic Lasers: Applications of Nonlinear Dynamics and Synchronization*, Wiley-VCH, Singapore, 2012.
- [9] J. Grzybowski, M. Eisencraft, and E. Macau, "Chaos-based communication systems: Current trends and challenges," in *Applications of Chaos and Nonlinear Dynamics in Engineering - Vol. 1* (S. Banerjee, M. Mitra, and L. Rondoni, eds.), vol. 71 of *Understanding Complex Systems*, pp. 203–230, New York: Springer-Verlag, 2011, doi: 10.1007/978-3-642-21922-1_7.
- [10] R. Candido, M. Eisencraft, and M. T. M. Silva, "Channel equalization for chaotic communications systems," in *Chaotic Signals in Digital Communications* (M. Eisencraft, R. Attux, and R. Suyama, eds.), CRC Press, Inc., 2013.
- [11] R. Candido, M. Eisencraft, and M. T. M. Silva, "Channel equalization for synchronization of chaotic maps," *Digital Signal Processing*, vol. 33, pp. 42–44, Oct. 2014, doi: 10.1109/TSP.2014.2333560.
- [12] R. Candido, D. C. Soriano, M. T. M. Silva, and M. Eisencraft, "Do chaos-based communication systems really transmit chaotic signals?," *Signal Processing*, vol. 108, no. 0, pp. 412 – 420, 2015, doi: 10.1016/j.sigpro.2014.10.004.
- [13] H.-P. Ren, C. Bai, J. Liu, M. S. Baptista, and C. Grebogi, "Experimental validation of wireless communication with chaos," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 26, p. 083117, aug 2016, doi: 10.1063/1.4960787.
- [14] R. Candido, M. T. M. Silva, and M. Eisencraft, "A new encoding and switching scheme for chaos-based communication," *Computational and Applied Mathematics*, pp. 1–14, 2017.
- [15] M. Eisencraft, J. V. C. Evangelista, R. A. Costa, R. T. Fontes, R. Candido, D. P. B. Chaves, C. Pimentel, and M. T. M. Silva, "New trends in chaos-based communications and signal processing," in *A Mathematical Modeling Approach from Nonlinear Dynamics to Complex Systems*, pp. 109–129, Springer International Publishing, jun 2018, doi: 10.1007/978-3-319-78512-7_7.
- [16] K. M. Cuomo and A. V. Oppenheim, "Communication using synchronized chaotic systems," US Patent, 5291555, 1994.
- [17] H. D. I. Abarbanel, N. F. Rulkov, L. S. Tsimring, and M. I. Rabinovich, "Chaotic communication apparatus and method for use with a wired or wireless transmission link," US Patent, 5923760A, 1996.
- [18] S.-s. Lee, Y.-H. Kim, and A. S. Dmitriev, "Method for direct chaotic communications with predetermined spectral mask," US Patent, 8116352B2, 2012.
- [19] A. J. Michaels and C. B. David, "Adaptive link communications using adaptive chaotic spread waveform," US Patent, 8369377B2, 2013.
- [20] N. J. Corron and J. N. Blakely, "Chaos in optimal communication waveforms," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, vol. 471, aug 2015, doi: 10.1098/rspa.2015.0222.
- [21] C. Williams, "Chaotic communications over radio channels," *IEEE Trans. Circuits Syst. I*, vol. 48, pp. 1394 –1404, Dec. 2001, doi: 10.1109/TCSI.2001.972846.
- [22] M. Eisencraft, R. D. Fanganiello, and L. H. A. Monteiro, "Chaotic synchronization in discrete-time systems connected by bandlimited channels," *IEEE Commun. Lett.*, vol. 15, pp. 671–673, Jun. 2011, doi: 10.1109/LCOMM.2011.040111.102309.
- [23] H. Leung, "System identification using chaos with application to equalization of a chaotic modulation system," *IEEE Trans. Circuits Syst. I*, vol. 45, pp. 314–320, Mar. 1998, doi: 10.1109/81.662721.
- [24] Z. Zhu and H. Leung, "Adaptive blind equalization for chaotic communication systems using extended-Kalman filter," *IEEE Trans. Circuits Syst. I*, vol. 48, no. 8, pp. 979–989, 2001, doi: 10.1109/81.940188.
- [25] J. Feng, C. K. Tse, and F. C. M. Lau, "Reconstruction of chaotic signals with application to channel equalization in chaos-based communication systems," *International Journal of Communication Systems*, vol. 17, no. 3, pp. 217–232, 2004, doi: 10.1002/dac.639.
- [26] M. Ciftci and D. Williams, "Iterative equalization for chaotic communications systems," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Process.*, vol. 4, pp. 165–68, 2005, doi: 10.1109/ICASSP.2005.1415971.
- [27] C. Vural and G. Çetinel, "Blind equalization of single-input single-output fir channels for chaotic communication systems," *Digital Signal Processing*, vol. 20, no. 1, pp. 201–211, 2010, doi: 10.1016/j.dsp.2009.06.001.
- [28] W. Liu, J. C. Príncipe, and S. Haykin, *Kernel Adaptive Filtering: A Comprehensive Introduction*. Wiley, 2010.

- [29] R. Candido, M. Eisencraft, and M. T. M. Silva, "Channel equalization for synchronization of Ikeda maps," in *Proc. of 21st European Signal Processing Conference (EUSIPCO'2013)*, (Marrakesh, Morocco), 2013.
- [30] C. W. Wu and L. O. Chua, "A simple way to synchronize chaotic systems with applications to secure communication systems," *International Journal of Bifurcation and Chaos*, vol. 3, pp. 1619–1627, Dec. 1993, doi: 10.1142/S0218127493001288.
- [31] M. Hénon, "A two-dimensional mapping with a strange attractor," *Communications in Mathematical Physics*, vol. 50, pp. 69–77, 1976, doi: 10.1007/BF01608556.
- [32] S. Haykin, *Adaptive systems*. New York: Wiley, 4th ed., 2000.
- [33] A. H. Sayed, *Adaptive Filters*. John Wiley & Sons, NJ, 2008.
- [34] W. D. Parreira, J. C. M. Bermudez, C. Richard, and J. Y. Tourneret, "Stochastic behavior analysis of the gaussian kernel least-mean-square algorithm," *IEEE Transactions on Signal Processing*, vol. 60, pp. 2208–2222, May 2012, doi: 10.1109/TSP.2012.2186132.
- [35] S. V. Vaerenbergh, L. A. Azpicueta-Ruiz, and D. Comminiello, "A split kernel adaptive filtering architecture for nonlinear acoustic echo cancellation," in *Proc. of the 24th European Signal Processing Conference (EUSIPCO)*, pp. 1768–1772, Aug. 2016, doi: 10.1109/EUSIPCO.2016.7760552.
- [36] M. Scarpiniti, D. Comminiello, R. Parisi, and A. Uncini, "A collaborative approach to time-series prediction," in *Neural Nets WIRN11* (B. Apolloni, S. Bassis, A. Esposito, and F. C. Morabito, eds.), pp. 178–185, IOS Press, 2011.
- [37] M. Yukawa, "Multikernel adaptive filtering," *IEEE Transactions on Signal Processing*, vol. 60, pp. 4672–4682, Sep. 2012, doi: 10.1109/TSP.2012.2200889.
- [38] F. A. Tobar, S. Y. Kung, and D. P. Mandic, "Multikernel least mean square algorithm," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, pp. 265–277, Feb. 2014, doi: 10.1109/TNNLS.2013.2272594.
- [39] W. Liu, P. P. Pokharel, and J. C. Principe, "The kernel least-mean-square algorithm," *IEEE Transactions on Signal Processing*, vol. 56, pp. 543–554, Feb 2008, doi: 10.1109/TSP.2007.907881.
- [40] A. H. Sayed, *Fundamentals of Adaptive Filtering*. John Wiley & Sons, NJ, 2003.
- [41] J. Platt, "A resource-allocating network for function interpolation," *Neural Computation*, vol. 3, pp. 213–225, June 1991, doi: 10.1162/neco.1991.3.2.213.
- [42] C. Richard, J. C. M. Bermudez, and P. Honeine, "Online prediction of time series data with kernels," *IEEE Transactions on Signal Processing*, vol. 57, pp. 1058–1067, Mar. 2009, doi: 10.1109/TSP.2008.2009895.
- [43] B. Chen, S. Zhao, P. Zhu, and J. C. Principe, "Quantized kernel least mean square algorithm," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, pp. 22–32, Jan 2012, doi: 10.1109/TNNLS.2011.2178446.
- [44] G. R. G. Lanckriet, N. Cristianini, P. Bartlett, L. E. Ghaoui, and M. I. Jordan, "Learning the kernel matrix with semidefinite programming," *Journal of Machine learning research*, vol. 5, no. Jan, pp. 27–72, 2004.
- [45] W. Gao, C. Richard, J. C. M. Bermudez, and J. Huang, "Convex combinations of kernel adaptive filters," in *Proc. of IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–5, Sep. 2014, doi: 10.1109/MLSP.2014.6958882.
- [46] D. Comminiello and J. C. Principe, eds., *Adaptive learning methods for nonlinear system modeling*. Oxford: Elsevier, 2018.
- [47] L. A. Azpicueta-Ruiz, J. Arenas-García, M. T. M. Silva, and R. Candido, *Adaptive Learning Methods for Nonlinear System Modeling*, ch. Combined filtering architectures for complex nonlinear systems, pp. 243–264. Oxford: Elsevier, 1st ed., 2018.
- [48] M. T. M. Silva, R. Candido, J. Arenas-García, and J. A. Azpicueta-Ruiz, "Improving multikernel adaptive filtering with selective bias," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Process.*, (Calgary, Canada), pp. 4529–4533, April 2018, doi: 10.1109/ICASSP.2018.8461290.
- [49] H. Fan, Q. Song, and S. B. Shrestha, "Kernel online learning with adaptive kernel width," *Neurocomputing*, vol. 175, pp. 233–242, 2016, doi: 10.1016/j.neucom.2015.10.055.
- [50] T. Wada and T. Tanaka, "Doubly adaptive kernel filtering," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 904–909, Dec 2017, doi: 10.1109/APSIPA.2017.8282173.
- [51] I. Cha and S. A. Kassam, "Channel equalization using adaptive complex radial basis function networks," *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 122–131, Jan 1995, doi: 10.1109/49.363139.
- [52] P. Bouboulis and S. Theodoridis, "The complex gaussian kernel LMS algorithm," in *Proceedings of the 20th International Conference on Artificial Neural Networks: Part II, ICANN'10*, (Berlin, Heidelberg), pp. 11–20, Springer-Verlag, 2010.



learning.

Renato Candido received the B.S. degree in 2006 from Universidade Presbiteriana Mackenzie, São Paulo, Brazil and the M.S. and Ph.D. degrees in 2009 and 2014 from Escola Politécnica, Universidade de São Paulo, Brazil, all in Electrical Engineering. From 2015 to 2017, he worked as a Postdoctoral Researcher at the Department of Electronic Systems Engineering, Escola Politécnica, Universidade de São Paulo and currently he collaborates as a researcher at the same university. His research interests include signal processing, adaptive filtering, and machine



works.

Marcio Eisencraft received the B.S. degree in 1998, the M.S. degree in 2001, and the Ph.D. degree in 2006, all in Electrical Engineering from Escola Politécnica, Universidade de São Paulo, Brazil. Since 2013, he has been with the Department of Telecommunications and Control Engineering at Escola Politécnica, Universidade de São Paulo, where he is currently an Associate Professor. His research interests include signal processing, digital communications, chaotic dynamics and applications, applied and computational mathematics, and complex net-



for the IEEE Signal Processing Letters. His research interests include linear and nonlinear adaptive filtering, and machine learning for signal processing.

Magno T. M. Silva received the B.S. degree in 1998, the M.S. degree in 2001, and the Ph.D. degree in 2005, all in Electrical Engineering from Escola Politécnica, Universidade de São Paulo, Brazil. Since August 2006, he has been with the Department of Electronic Systems Engineering at Escola Politécnica, Universidade de São Paulo, where he is currently an Associate Professor. From January to July 2012, he worked as a Postdoctoral Researcher at Universidad Carlos III de Madrid, Leganés, Spain. From 2015 to 2018, he served as Associate Editor