A Novel Kernel Algorithm for Finite Impulse Response Channel Identification

Rachid Fateh[1], Anouar Darif[1], Ahmed Boumezzough[2], Said Safi[1], and Miloud Frikel[3]

[1]*Laboratory of Innovation in Mathematics, Applications and Information Technologies (LIMATI), Polydisciplinary Faculty, Sultan Moulay Slimane University, Beni Mellal, Morocco,*
[2]*Laboratory of Research in Physics and Engineering Sciences, Polydisciplinary Faculty, Sultan Moulay Slimane University, Beni Mellal, Morocco,*
[3]*Laboratoire d'Automatique de Caen, UNICAEN, ENSICAEN, Normandie University, Caen, France*

Abstract — Over the last few years, kernel adaptive filters have gained in importance as the kernel trick started to be used in classic linear adaptive filters in order to address various regression and time-series prediction issues in nonlinear environments. In this paper, we study a recursive method for identifying finite impulse response (FIR) nonlinear systems based on binary-value observation systems. We also apply the kernel trick to the recursive projection (RP) algorithm, yielding a novel recursive algorithm based on a positive definite kernel. For purposes, our approach is compared with the recursive projection (RP) algorithm in the process of identifying the parameters of two channels, with the first of them being a frequency-selective fading channel, called a broadband radio access network (BRAN B) channel, and the other being a a theoretical frequency-selective channel, known as the Macchi channel. Monte Carlo simulation results are presented to show the performance of the proposed algorithm.

Keywords — finite impulse response, kernel adaptive filtering, kernel recursive projection identification, nonlinear system identification.

1. Introduction

Linear adaptive filters are a class of digital filters that can automatically adjust their parameters based on the input data. They are commonly used for signal processing tasks, such as noise reduction, echo cancellation, and equalization [1]–[4]. The basic idea behind linear adaptive filters is to use an algorithm that updates the filter coefficients in response to changes in the input signal [5], [6]. The most widely used algorithm for this purpose is the least mean squares (LMS) algorithm [7], which iteratively adjusts the filter coefficients to minimize the mean squared error between the filter's output and the desired output. The performance of linear adaptive filters depends on several factors, including the choice of filter structure, the algorithm applied to update the filter coefficients and the design of the input signal. In general, these filters work best when the input signal is stationary or varies slowly over time and when the filter structure is chosen to match the statistical properties of the signal.

One of the most challenging problems encountered in engineering digital communication systems consists in minimizing the impact of the communication channel. The most common solution to this issue consists in estimating the channel's impulse response parameter, and then using an equalizer to equalize it (see e.g. [8]–[12]). This approach relies strongly on the quality of the estimation, also known as system identification. System identification (SI) is an area of key interest in the field of automatic control. In SI, the aim is to build the most adequate mathematical models of dynamic systems based on experimental data, i.e., using measurements of the system input/output (IO) signals [13]–[16].

At present, binary-valued observation systems are receiving a great deal of attention [17]–[22] thanks to the extensive applications of binary-valued sensors, including asynchronous transfer mode (ATM), Hall-effect sensors for velocity and acceleration, witching sensorsfor exhaust-gas oxygen level, industrial sensors for brushless DC motors, photoelectric position sensors, etc. [23], [24]. The input (or output) monitored in these systems cannot be directly measured. Instead, it is the information whether the input (or output) of the system is superior or inferior to a specific numerical value, called a threshold (a key factor for binary-valued systems), that can be measured and used to implement a system controller. In terms of binary-valued observation systems, there are several important results impacting the recursive identification of single-input single-output (SISO) finite impulse response (FIR) communication channels, the identification of infinite impulse response (IIR) systems, and state estimation problems that are presented in [25]–[28].

Over the last few years, a technique based on kernel adaptive filtering (KAF) [29], [30] has been employed in a wide range of telecommunication applications. KAF is proposed as a way to overcome the limitations of classic linear filtering techniques when handling non-linear systems.

Linear filtering techniques, such as the LMS algorithm and its normalized variant (NLMS algorithm) [5], suppose that the input signal and the filter coefficients have a linear relationship, which may not be accurate in many real-world applications. KAF integrates kernel methods which can map data to a higher-dimensional space where classical linear techniques can be more accurate, with adaptive filtering techniques to handle non-linearity and non-stationarity in the input signal, where linear filtering can be executed [31]. The kernel function can be chosen to correspond to the input sig-

nal's properties, enabling KAF to adapt to the change in the signal's properties over time.

One of the key advantages of kernel-based methods is that they can work with non-linearly separable data. They are based on the principle that a decision boundary in a high-dimensional reproducing kernel Hilbert space (RKHS) [32] can be represented as a linear boundary in a lower-dimensional space. This allows kernel methods to effectively handle complex, nonlinear relationships between the input features and the output variable, which represents an emerging technique for machine learning applications, i.e. regularization networks [33], Gaussian process regression (GPR) [34], and support vector machines (SVMs) [35], as well as for nonlinear signal processing and classification.

At present, there are many adaptive kernel filtering algorithms that have been described in the literature of the subject. Some of them include the following: kernel affine projection algorithms (KAPA) [36], kernel principal component analysis (KPCA) [37], kernel least mean squares (KLMS) [38], and kernel recursive least square (KRLS) [39]. To increase the robustness of these adaptive kernel filtering algorithms, many different variants have been developed, such as quantized kernel recursive least squares (QKRLS) [40], quantized kernel least mean square (QKLMS) [41], extended kernel recursive least squares (Ex-KRLS) [42], kernel least mean square with adaptive kernel size (KLMS-AKS) [43], random Fourier feature kernel recursive least squares (RFF-KRLS) [44], quantized kernel least lncosh (QKLL) [45], and kernel extended improved proportionate normalized least mean square algorithm (KE-IPNLMS) [46]. In this paper, we address the recursive identification problem of finite impulse response (FIR) single-input single-output (SISO) communication channels using kernel techniques. Indeed, the method proposed by Guo and Zhao in [26] for linear systems with binary-valued observations is extended to the general case of nonlinear system identification using kernel methods. Several simulation results, both under noisy environments and for diverse data lengths N, are provided to demonstrate the accuracy of the proposed kernel method. Our proposed algorithm is referred to as the KRPI algorithm.

The paper is organized as follows. In Section 2, we described the configuration of the nonlinear system identification problem based on binary-valued output observations. In Section 3, we briefly presented the recursive projection (RP) algorithm. In Section 4, we examined some fundamental notions of the kernel approaches. In addition, the proposed kernel recursive projection identification (KRPI) algorithm for nonlinear systems with binary output measurements is explained in Section 5. Some simulations to demonstrate the effectiveness of the proposed algorithm are shown in Section 6. Finally, Section 7 focuses on a brief discussion and concludes the paper.

2. System Descriptions and Assumptions

We consider the single-input single-output (SISO) nonlinear system (presented in Fig. 1) described by the following Eq. (1):
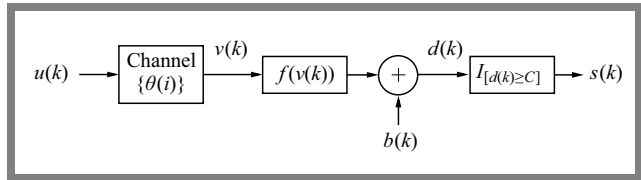

Fig. 1. System configuration.

$$\begin{cases} v(k) = \sum_{i=0}^{L-1} \theta(i)u(k-i), \\ d(k) = f\big(v(k)\big) + b(k), \quad k = 0, 1, 2, \ldots, N \end{cases} \quad (1)$$

where $u(n) = [u(k), u(k-1), \ldots, u(k-L+1)]^\top \in \mathbb{R}^L$, $\theta = [\theta(0), \theta(1), \ldots, \theta(L-1)]^\top \in \mathbb{R}^L$ and $f(.)$ are the input sequence, channel coefficient (parameter vector), of size L, and non-linearity, respectively. The transpose operator is denoted by the superscript $\top$. $d(k)$ is the system output corrupted by the measurement noise $b(k)$. Using a binary detector $I[.]$ equipped with a predefined threshold $C \in \mathbb{R}$, the output of the system $d(k)$ becomes measurable. The quantized output data $s(k)$ could be expressed by:

$$s(k) = I_{[d(n) \geqslant C]} = \begin{cases} 1 & \text{if } d(n) \geqslant C \\ -1 & \text{otherwise}. \end{cases} \quad (2)$$

To make the system analysis easier and to come up with a good result, we assume, in this paper, that:

Assumption 1. The input sequence $\{u(n)\}$ is an i.i.d. (independent and identically distributed) bounded random process with zero mean.

Assumption 2. At any time k, input $\{u(n)\}_{k=1}^N$ satisfies the following equation:

$$\theta^\top u(n) \neq 0. \quad (3)$$

Assumption 3. The additive noise $\{b(k)\}$ is Gaussian, i.i.d. and independent of $\{u(k)\}$ and $\{d(k)\}$ (bounded).

Assumption 4. The nonlinear function $f(.)$ is invertible and continuous for all finite u.

Assumption 5. There is no delay in the system, i.e. $\theta(0) \neq 0$.

Assumption 6. The C-value is available (i.e. known).

Our primary objective in this paper is to construct a recursive kernel identification algorithm for FIR systems by using positive definite kernels and binary observations $s(k)$, in order to provide a recursive estimate of θ.

3. Recursive Projection (RP) Algorithm

A recursive projection algorithm for finite impulse response (FIR) system identification with binary-valued observations under sufficiently rich inputs and a fixed quantized threshold was presented by Guo and Zhao [26]. In this paper, we briefly present this algorithm. The goal of this method is to estimate the vector parameter θ in real time. It is based on the following assumptions:

Assumption 1. $\theta \in \Theta \subset \mathbb{R}^L$, where Θ is a bounded convex compact set.

Assumption 2. The measurement noise $\{b(k),\ k \geqslant 1\}$ is an i.i.d. sequence of random variables with finite variance σ^2 and zero mean. $F(.)$ and $f(.)$ are the distribution and density functions of $b(1)$, respectively. Note that $F(.)$ and $f(.)$ are assumed to be known.

Assumption 3. Input sequence $u(k)$ is $F(k-1)$-measurable with $F(k-1)$ being the σ-algebra that is generated by $b(k), \ldots, b(k-1)$.

The update equations of the RP algorithm are given by:

$$\hat{\theta}(k) = \sqcap_{\Theta} \left\{ \hat{\theta}(k-1) + \frac{u(k)}{r(k)} \right\}, \qquad (4)$$

$$r(k) = 1 + \sum_{i=1}^{k} u^{\top}(i)u(i), \qquad (5)$$

$$\hat{s}(k) = \beta \left(F \left(C - u^{\top}(k)\hat{\theta}(k-1) \right) - s(k) \right), \qquad (6)$$

where:

– $\hat{\theta}(k)$ represents the estimation of $\theta(k)$ at time k,

– $\sqcap_{\Omega}$ is the projection operator in a convex compact set $\Omega \subseteq \mathbb{R}^L$, which is defined as:

$$\sqcap_{\Omega} = \arg \min_{\omega \in \Omega} \|u - \omega\|, \forall u \in \mathbb{R}^L, \qquad (7)$$

– $\beta > 0$ denotes the constant scalar, that has an important role in the convergence of $\hat{\theta}(k)$.

From the point of view of search procedures, new challenges arise and the recursive projection algorithm does not work due to the difficult knowledge of the noise distribution function (when the channel error is unbounded). Therefore, many extensions of this algorithm are proposed, see for example [25], [47], [48].

4. Kernel-based Adaptive Filters

Kernel methods are a class of machine learning algorithms that are widely used for classification, regression, identification, and other tasks [29], [30], [32]. They operate by transforming data into a higher-dimensional space in which it can be more easily separated.

One of the key advantages of kernel methods is that they can work with non-linearly separable data. They are based on the principle that a decision boundary in a high-dimensional space can be represented as a linear boundary in a lower-dimensional space. This allows kernel methods to effectively handle complex, nonlinear relationships between the input features and the output variable.

The present section will proceed by presenting a short background on the crucial idea behind KAF techniques. The reproducing kernel theory [49] has allowed many adaptive algorithms to evolve. The suitability of these approaches for training is founded on the concept of learning by error correction. For the purpose of implementing the kernel methods over a measurement data input $\mathcal{U}$, we only need to obtain kernel function values for each pair of the measurement data input. Usually, these values are stored in a square matrix named the kernel matrix (or the Gram matrix).

Definition 1. The Gram matrix, also known as the kernel matrix, is a matrix that summarizes the pairwise similarities between a set of data points using a kernel function. Specifically, given a set of N input data points $u_1, u_2, \ldots, u_N$, the Gram matrix K is an $N \times N$ matrix whose (i, j)-th entry is defined as:

$$K_{ij} = \kappa(u_i, u_j), \text{ for } i, j = 1, \ldots, N, \qquad (8)$$

where κ is the kernel function.

The Gram matrix is important in kernel methods, such as kernelized support vector machines, as it allows us to compute the dot products of data points in the feature space implicitly defined by the kernel function, without explicitly computing the feature vectors themselves. This is because the dot product of two feature vectors in the feature space can be expressed as a function of the corresponding entries of the Gram matrix.

In the remainder of the document we denote, by κ, the reproducing kernel and by $\mathcal{U}$ the input space. Like the inner product, we can also expect the kernel function to be positive definite.

Definition 2. A positive definite kernel is a function that takes pairs of inputs and produces a measure of their similarity. More formally, a kernel function κ is positive definite if for any finite set of inputs $u_1, u_2, \ldots, u_N$, the corresponding kernel matrix $K_{ij} = \kappa(u_i, u_j)$ is positive definite, meaning that its eigenvalues are all non-negative:

$$\sum_{i,j=1}^{N} a_i a_j \kappa(u_i, u_j) \geqslant 0, \qquad (9)$$

for all $N \in \mathbb{N}$, $(u_1, \ldots, u_N) \subseteq \mathcal{U}^N$ and $(a_1, \ldots, a_N) \subseteq \mathbb{R}^N$.

In machine learning, positive definite kernels are commonly used in kernel methods, such as support vector machines and kernelized ridge regression [50]. They allow us to implicitly map the input data into a high-dimensional feature space, where linear methods can be applied even when the data is not linearly separable. Positive definite kernels have a number of desirable properties, including being symmetric and positive semi-definite, and they can be used to define a notion of distance between data points that takes into account their similarity in the feature space.

Common examples of positive definite kernels include the linear kernel $\kappa(u, u') = (\langle u, u' \rangle) = u^{\top} u'$, the polynomial kernel $\kappa(u, u') = (\langle u, u' \rangle + c)^p$, the sigmoidal kernel $\kappa(u, u') = \text{tgh}(a \langle u, u' \rangle + c)$ and others.

Symmetric and positive definite scalar functions, often referred to simply as "kernel", are more precisely "Mercer kernel". This expression comes from what is called "Mercer's theorem".

Theorem 1. Mercer's theorem is a fundamental result in kernel methods which provides the necessary and sufficient

condition for a function to be a valid positive definite kernel function. Specifically, Mercer's theorem states that a continuous and symmetric function $\kappa(u, u')$ on a compact domain is a valid positive definite kernel function if and only if it can be expressed as:

$$\kappa(u, u') = \sum_{i=1}^{\infty} \zeta_i \Phi_i(u) \Phi_i(u') \,, \tag{10}$$

where ζ_i and Φ_i are the eigenvalues and eigenfunctions of the integral operator $T_\kappa : \mathcal{L}^2(\mathcal{U}) \to \mathcal{L}^2(\mathcal{U})$, defined by:

$$T_\kappa(f)(.) = \int_{\mathcal{U}} \kappa(u, .) f(u) \mathrm{d}u \tag{11}$$

and the series converges absolutely and uniformly.

In simpler terms, Mercer's theorem says that a function $\kappa(u, u')$ is a valid positive definite kernel function if and only if it can be written as a weighted sum of inner products of feature functions $\Phi_i(u)$ and $\Phi_i(u')$, with non-negative weights ζ_i.

Mercer's theorem is important in kernel methods because it provides a way to construct valid kernel functions for a given problem, by finding a suitable set of feature functions and their corresponding weights that satisfy the conditions of the theorem. It also provides a theoretical foundation for the effectiveness of kernel methods, by showing that any positive definite kernel function can be used to implicitly map the input data into a high-dimensional feature space, where linear methods can be applied effectively even when the data is not linearly separable in the original space.

In reproducing kernel Hilbert space, the mapping representation Φ_i would be made as follows:

$$\Phi(x) = \left[\sqrt{\zeta_1} \Phi_1(u), \sqrt{\zeta_2} \Phi_2(u), \ldots \right]^\top \,. \tag{12}$$

As a means of representing the components of $\mathcal{H}$ from their coordinates, a Hilbert space $\mathcal{H}$ must be given an orthonormal base. The assigned kernel of this should be a symmetric, continuous, positive definite function $\kappa : \mathcal{U} \times \mathcal{U} \longmapsto \mathbb{R}$, normalized, where $\mathcal{U} \subset \mathbb{R}^N$ is a compacted subset.

5. Kernel Recursive Projection Identification (KRPI)

In this section, we will introduce the proposed algorithm. The general concept is to operate the algorithm introduced in [26] in the feature space of the kernel that connects to a positive definite kernel κ, under the characteristic mapping $\Phi(.)$ as defined in Eq. (13). The sample sequence is transformed by using a feature map Φ:

$$\Phi \ : \ \mathcal{U} \longrightarrow \mathcal{H}$$
$$u(i) \longrightarrow \kappa(u(i), .), \ 0 \leqslant i \leqslant N. \tag{13}$$

In order to establish the model of Hilbert spaces with a reproducing kernel, we have decided to use the (Gaussian) radial basis function (RBF) kernel that is, ideally, a predefined se-

lection due to its general-purpose property of approximation and numeric stability. The reason for choosing the Gaussian kernel function is that it is a convex optimization problem. We have done a convex subset on the solution provider space while using a Gaussian function, and we establish a norm on a vector space when we use the Mercer kernel. The optimization problem has almost one optimal point, since it is a strictly convex Gaussian kernel:

$$\kappa(u, u') = \mathrm{e}^{-\frac{\|u - u'\|^2}{2\sigma^2}} \,, \quad \forall (u, u') \in \mathcal{U}^2 \,, \tag{14}$$

where $\sigma > 0$ represents the smoothing parameter.

Figure 2 illustrates the mapping of the data space $\mathcal{U}$ to the Hilbert space $\mathcal{H}$ obtained by the Gaussian reproducing kernel κ.
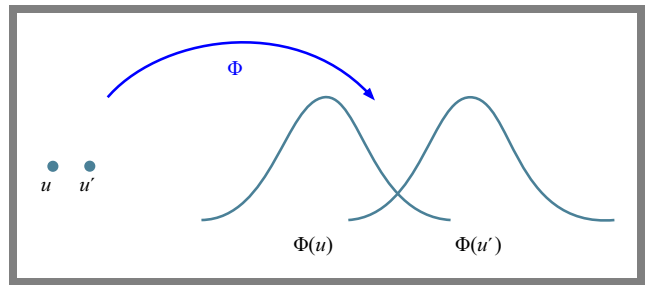

Fig. 2. Define a characteristic map.

The block diagram shown in Fig. 3 illustrates the adaptive kernel-based channel estimation using the proposed algorithm. The system inputs and outputs, $u(k)$, $s(k)$ are considered measurable.
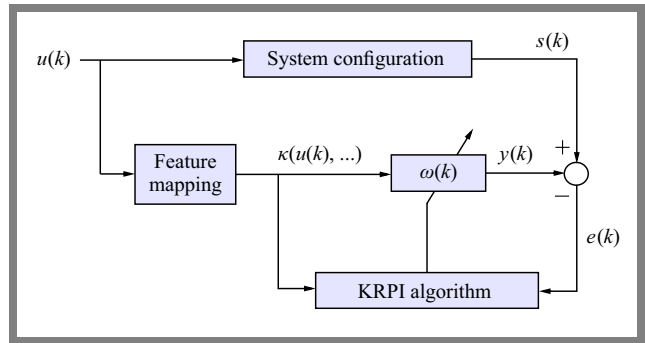

Fig. 3. Schematic of kernel adaptive system identification.

The proposed KRPI algorithm is defined by the following steps:
– Step 1. Transformation of measured data. As the initial step, in order to create input data, the observation data space $\mathcal{X}$ is transformed into a nonlinear Hilbert space $\mathcal{H}$:

$$\left\{ (\Phi(u(1)), \ s(1)), \ldots, (\Phi(u(N)), \ s(N)), \ldots \right\}. \tag{15}$$

– Step 2. Application of the RP algorithm methodology. The Guo *et al*. algorithm is used in the second step to minimize the cost function by applying its logic to the data sequence generated in the first step that is formulated in Eq. (15):

$$E\left[|s(k) - \langle (\Phi(u(k))), \omega \rangle_{\mathcal{H}} |^2 \right], \tag{16}$$

where the term ω represents the weight vector in reproducing kernel Hilbert space $\mathcal{H}$.

– Step 3. Determination of the reproducing kernel in Hilbert space. In this step, we proceed directly in the feature space $\mathcal{H}$, assuming that our data have already been successfully modeled in RKHS using the mapping function Φ, i.e.:

$$\mathcal{X} \ni x \longrightarrow \Phi(x(n)) := \kappa(x,.) \in \mathcal{H} . \tag{17}$$

The kernel recursive projection identification algorithm for estimating the parameter vector ω is summarized in Algorithm 1.

Algorithm 1. Kernel recursive projection identification (KRPI).

Input: Samples $\{u(k), s(k)\}$, $k = 1, 2, \ldots, N$.

Initialization:

Initialize weight vector $\omega(0)$.

Select the kernel bandwidth σ, constant scalar β, data length N, and the threshold C.

Computation:

While $\{u(k), s(k)\}_{k=1}^{N}$ available **do**:

1) Compute $\Phi(k)$ as:
   $\Phi(k) = \left[\kappa(u(k), u(1)), \ldots, \kappa(u(k), u(k-1)) \right]^{\top}$

2) Compute system's output:
   $y(k) = \omega(k)^{\top} \Phi(k)$

3) Compute the prediction error as:
   $e(k) = s(k) - y(k)$

4) Compute $r(k)$ as:
   $r(k) = 1 + \sum_{i=1}^{k} \Phi^{\top}(i) \Phi(i)$

5) Compute the estimation of $s(k)$ as:
   $\hat{s}(k) = \beta \left(F \left(C - \Phi^{\top}(k) \hat{\omega}(k-1) \right) - s(k) \right)$

6) Update weight vector:
   $\omega(k) = \sqcap_{\omega} \left\{ \hat{\omega}(k-1) + \dfrac{\Phi(k)}{r(k)} \right\}$

End while

6. Simulations

We proceed in this paragraph with Monte Carlo simulations to investigate the performance of the proposed KRPI algorithm in nonlinear system identification (NSI) with binary-valued output observations (Fig. 1) under different data inputs. We compare the performance of KRPI with that of RP in finite impulse response identification. Their parameters are set as: KRPI ($C = 0.5$, $\beta = 1000$, $\sigma = 0.5$), and RP ($C = 0.5$, $\beta = 1000$). It is worth noting that if we change one of these simulation parameters, the others, remain unchanged. The mean square error (MSE) of the estimated impulse response parameters is chosen as a specific metric to evaluate the performance of these algorithms, which is defined here as follows:

$$\text{MSE} = \frac{1}{L} \sum_{i=1}^{L} \left[\frac{\omega(i) - \hat{\omega}(i)}{\omega(i)} \right]^{2} , \tag{18}$$

where $\omega(i)$ and $\hat{\omega}(i)$) represent the measured and estimated impulse response parameters in each iteration, respectively.

All simulations were implemented using Matlab software. To examine the strength of the measurement noise for each algorithm, we define the signal-to-noise ratio (SNR) by the following relationship:

$$SNR = 10 \log_{10} \left[\frac{E\left(v^2(n)\right)}{E\left(b^2(n)\right)} \right] , \tag{19}$$

where $E[.]$ is the mathematical expectation.

For all simulations, we considered the function $f(.)$ (see Fig. 1) as a hyperbolic tangent function (tgh). It is an indefinitely differentiable function that realizes a bijection of $\mathbb{R}$ on $[-1, 1]$:

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} . \tag{20}$$

Since we have a non-linear system with binary output $s(k) = 1$ or -1, it corresponds to the asymptotes of the hyperbolic tangent function:

– $\lim_{x \to +\infty} \text{tgh}(x) = 1$,
– $\lim_{x \to -\infty} \text{tgh}(x) = -1$.

The physical interpretation of the hyperbolic tangent function in this context is that it introduces a nonlinear relationship between the input and output signals of the system.

The Figs. 4 and 5 show the input and output signals of the nonlinear system identification with binary-valued output observations (Fig. 1). The bottom graphs show the complete signal form of data length $N = 1000$ and the top graphs are zoomed in to between 400 to 500 data lengths to give more accurate details of the processed signals.
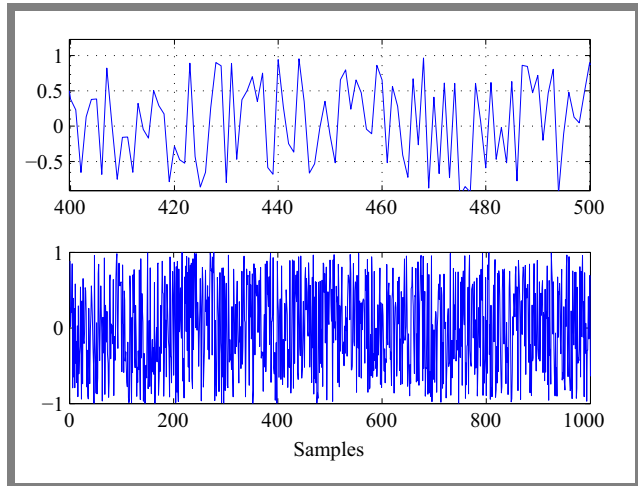
Fig. 4. Input sequence: top: zoomed-in, middle: 400 and 500 samples, bottom: full 1000 samples.

6.1. ETSI BRAN Mobile Radio Channels

The broadband radio access networks (BRAN) channel is a type of a wireless channel that is used in broadband wireless communication systems, such as WiMAX and 4G LTE. The BRAN channel is characterized by its wide
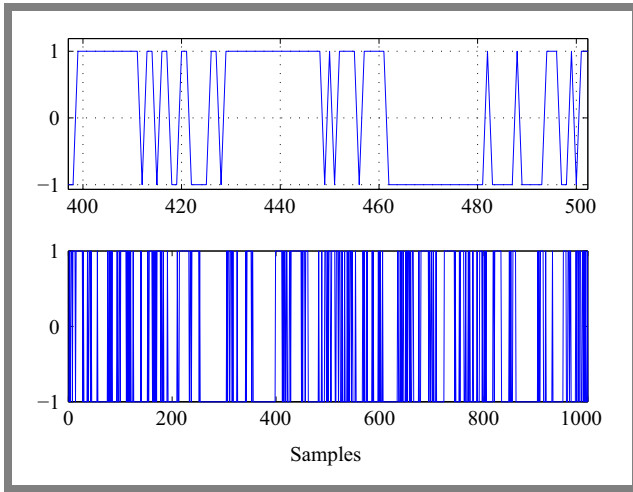
Fig. 5. Example of binary-valued output data: top zoomed-in to 400–500 samples, bottom full 1000 samples.

bandwidth, typically spanning several hundred megahertz, and its high data rates, which can be several tens of megabits per second.

In this subsection, we apply the algorithms presented in NSI to assess the benefit of the proposed KRPI algorithm in identifying the parameters of ETSI BRAN mobile radio channels: BRAN B. This model is suggested for use in large open areas and typical indoor scenarios with considerable delays under non-line-of-sight (NLOS) propagation settings. The details about the ETSI BRAN mobile radio channels can be found in [51], [52].

The ETSI BRAN radio channels' impulse response is described by:

$$h(n) = \sum_{i=0}^{L-1} \omega_i \delta(n - \tau_i), \ L = 18 \ , \qquad (21)$$

where $\delta(n)$, τ_i and $\omega_i \in N(0,1)$ denote, the Dirac function, the path i time delay and path i magnitude, respectively.

Table 1 summarizes the measured impulse response of the ETSI BRAN B radio channel.

Tab. 1. Delay and magnitudes of 18 targets of ETSI BRAN B radio channel.

Del. τ_i[ns]	Mag. ω_i[dB]	Del. τ_i[ns]	Mag. ω_i[dB]
0	-2.6	230	-5.6
10	-3.0	280	-7.7
20	-3.5	330	-9.9
30	-3.9	380	-12.1
50	0	430	-14.3
80	-1.3	490	-15.4
110	-2.6	560	-18.4
140	-3.9	640	-20.7
180	-3.4	730	-24.6

### 6.1.1. Impulse Response Parameter Estimation

In Fig. 6, for a data length of N fixed at 3000, with $SNR = 20$ dB and 50 Monte Carlo iterations, we have plotted the parameters of ETSI BRAN B channel estimated as a function of the path delays using the two algorithms. From this figure, we noticed that the proposed KRPI algorithm gives the best performance because the estimated parameters of the ETSI BRAN B channel impulse response are closer to the true one. Meanwhile, for the RP algorithm, we have a difference between the estimated and measured impulse response.
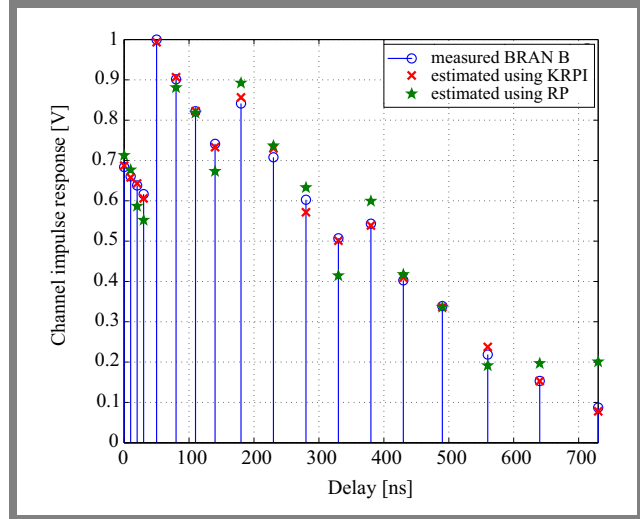
Fig. 6. Estimation of the ETSI BRAN B channel impulse response for a data length of $N = 3000$ and $SNR = 20$ dB.
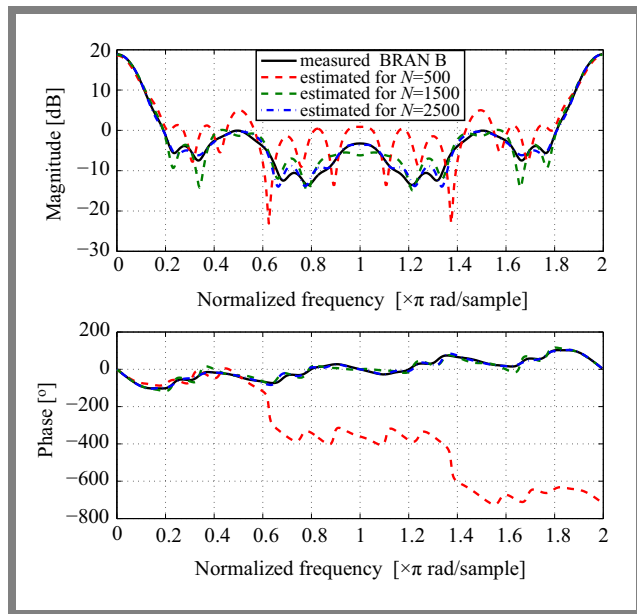
6.1.2. Magnitude and Phase Estimation

Using the proposed kernel recursive projection identification (KRPI) and recursive projection (RP) algorithms, we have identified, in this paragraph, the amplitude and phase of the
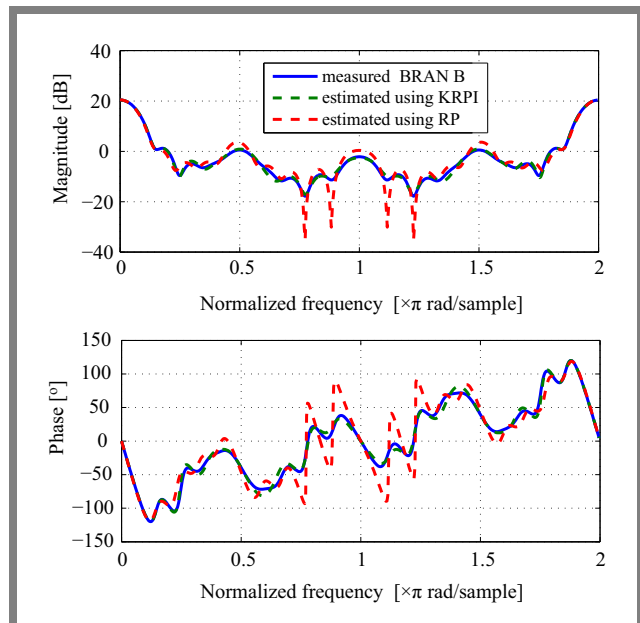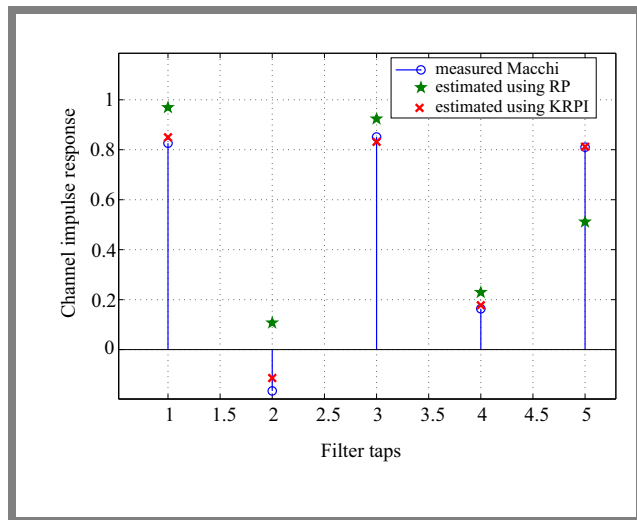

Fig. 7. Estimation of the amplitude and phase of the ETSI BRAN B, for a data length $N = 3000$ and $SNR = 20$ dB.

ETSI BRAN B channel impulse response. Figure 7 represents the estimation of the ETSI BRAN B magnitude and phase for an $SNR = 20$ dB and a data length of $N = 3000$. As shown in this figure, the proposed algorithm offers a very accurate estimation of the amplitude and phase compared to the RP algorithm. Also, in the same figure (Fig. 7), we have more parameters which are the same as those measured for ETSI BRAN B in the case when we apply the proposed KRPI algorithm. The evident difference between the estimated amplitude and phase patterns and the real model is clearly seen when using the RP algorithm.

Fig. 8. Estimation of the amplitude and phase of the ETSI BRAN B for different data lengths N and $SNR = 20$ dB.

Fig. 9. Estimation of the Macchi channel impulse response for a data length of $N = 3000$ and $SNR = 20$ dB.

Figure 8 illustrates the results of the estimation of the ETSI BRAN B magnitude and phase by employing the proposed KRPI algorithm for a fixed signal-to-noise ratio SNR of 20 dB and various data inputs N. Figure 9 demonstrates

that for the data length of $N = 2500$, estimations of the ETSI BRAN B magnitude and phase are very close to the true ones and they remain in complete agreement. Regarding the impulse response of ETSI BRAN B channel for one data length of $N = 1500$ the estimated phase of the ETSI BRAN B channel is closer to the true value, but the estimated magnitude follows the real model with minor differences. In unfavorable conditions, i.e. small data sample ($N = 500$), the performance of the proposed algorithm degrades during magnitude and phase estimation. To summarize, the data length N has a significant impact on the estimated phase, but only a minor impact on the estimated magnitude. The proposed algorithm needs, therefore, a large data sample to obtain an accurate approximation of the real values.

6.2. Macchi Channel

We adopted the Macchi channel to evaluate the theoretical outcome of the proposed algorithm. The Macchi channel, also known as the Macchi-Hall channel, is a quantum communication channel that models the transmission of quantum information between two parties in the presence of a noisy environment [53]. The impulse response of this channel is described by the following model:

$$\begin{cases} v(k) = 0.8264u(k) - 0.1653u(k-1) + 0.8512u(k-2) \\ \quad\quad + 0.1636u(k-3) + 0.81u(k-4), \\ \text{Zeros: } z_1 = 0.5500 + 0.9526i, \ z_2 = 0.5500 - 0.9526i, \\ \quad\quad z_3 = -0.4500 + 0.7794i, \ z_4 = -0.4500 - 0.7794i. \end{cases}$$

In the Macchi channel, the sender prepares a qubit in a particular state and then transmits it to the receiver through a noisy quantum channel that introduces errors and disturbances to the qubit state. The noisy channel is modeled as a sequence of independent and identically distributed (i.i.d.) quantum operations, with each of them being capable of causing a certain type of error.

6.2.1. Impulse Response Parameter Estimation

Under the conditions of $SNR = 20$ dB and $N = 3000$, Fig. 9 illustrates the estimations of Macchi's channel impulse response parameters using the two algorithms presented previously. This comparison proves that the proposed KRPI algorithm can be successfully used for the identification of the Macchi channel impulse response parameters. The average estimation values are close to the real model. This could be due to the nonlinear nature of the system which we need to identify, and the proposed KRPI algorithm effectively employs a linear model in a high-dimensional nonlinear space, which is equivalent to applying a nonlinear technique in the original space. In the opposite situation, the RP algorithm is a linear method.
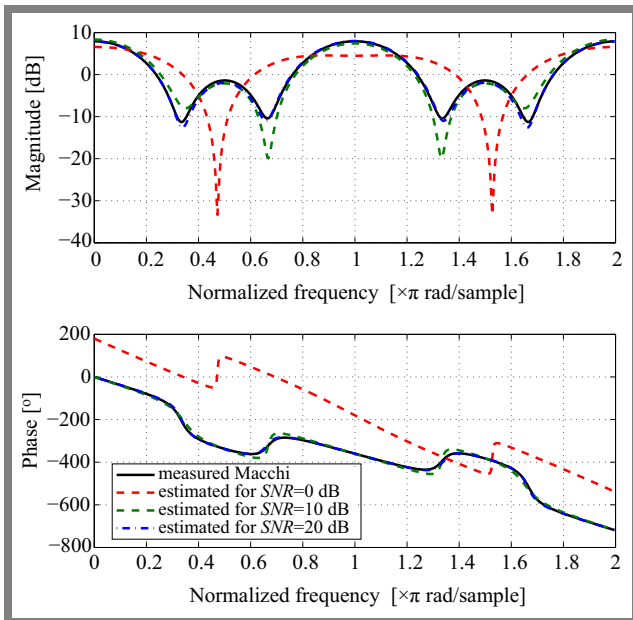
6.2.2. Magnitude and Phase Estimation

In this subsection, the Macchi channel impulse response magnitude and phase will be estimated. Figure 10 repre-

sents the estimation of the Macchi channel's magnitude and phase using the KRPI and RP algorithms for a data length of $N = 3000$ and for an $SNR = 20$ dB. Based on this result, we can conclude that, in comparison to the RP algorithm, the proposed KRPI algorithm is very important, since it offers approximately the same amplitude and phase (estimated and measured). Using the RP algorithm, we obtain a large difference between the estimated and measured phases.
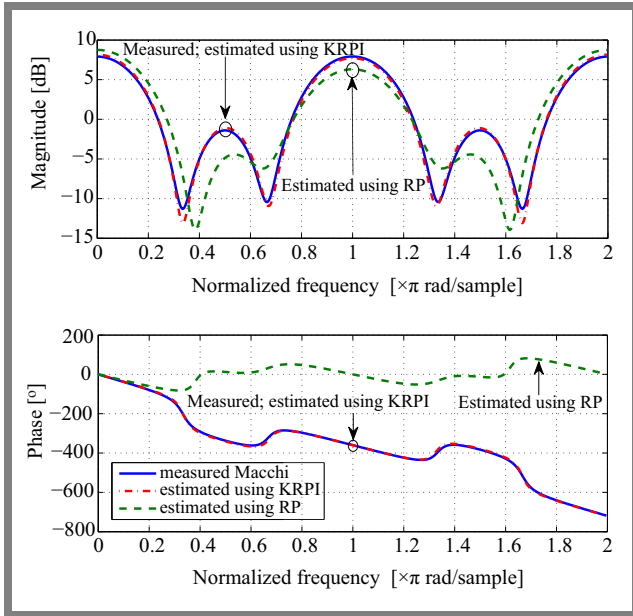
Fig. 10. Estimation of the amplitude and phase of the Macchi channel for a data length of $N = 3000$ and $SNR = 20$ dB.

Fig. 11. Estimation of the amplitude and phase of the Macchi channel for different SNR and $N = 3000$.

Now, we want to see how much the noise level affects the proposed KRPI algorithm's behavior. To do so, we run a simulation with various SNRs by adjusting the value of N.

Figure 11 shows the average values of the estimated parameters using the proposed KRPI algorithm. We can see that, in a very noisy environment (lowest $SNR = 0$ dB), the Gaussian noise influenced the estimated parameters of the model, as we see a significant difference between the estimated values of the amplitude and phase and the measured data, and a slight influence of the noise in the estimation of the parameters of the impulse response, principally if the variance of the noise is small (e.g. when we take $SNR = 10$ dB). If $SNR = 20$ dB, the estimated magnitude and phase are very close to the measurement and in perfect accordance. According to this result, we can notice that the performance of the proposed KRPI algorithm in a very noisy environment (SNR tends towards 0 dB) is acceptable compared to the RP algorithm.

6.3. Mean Square Error Criterion

In this subsection, the performance of the KRPI and RP methods is compared, where SNR ranges from 0 dB to 30 dB. The mean square error (MSE) results are reported in Tables 2 and 3. As presented in these tables, the use of both presented algorithms shows that MSE decreases if SNR increases for a fixed N. For the two studied channels (Macchi and ETSI BRAN B), the proposed KRPI algorithm ensures better performance convergence than the RP algorithm for all SNR values, even in a high noise environment ($SNR = 0$ dB), due to the fact that MSE values of the proposed KRPI are much lower than those obtained by the RP algorithm. For example, in the case of the ETSI BRAN B impulse response channel, the MSE value achieved by the proposed KRPI algorithm amounts to only 9.44% and 6.6% of the MSE value obtained using the RP algorithm, when the SNR is equal to 0 dB and 10 dB, respectively, and represents 20.56% and 0.68% of the MSE value obtained by the RP algorithm in the case of the Macchi impulse response channel for $SNR = 20$ dB and 30 dB, respectively.

Tab. 2. Comparison of performance of KRPI and RP for different SNR values and for a data length of $N = 3000$ in the case of the ETSI BRAN B channel.

SNR [dB]	MSE: RP	MSE: KRPI
0	2.7917	0.2636
10	2.3604	0.1577
20	0.1023	0.0833
30	0.0568	0.0099

Tab. 3. Comparison of performance of KRPI and RP for different SNR values and for a data length of $N = 3000$ in the case of the Macchi channel.

SNR [dB]	MSE: RP	MSE: KRPI
0	0.4855	0.3465
10	0.0288	0.0238
20	0.0214	0.0044
30	0.0878	0.0006

7. Conclusion and Future Work

In this paper, we proposed a novel kernel method for single-input single-output (SISO) nonlinear system identification with binary-valued observations. This approach employs a kernel function to perform an implicit mapping of the data using the kernel trick. This kernel trick-based approach implicitly maps the original measured data input into an infinite (or high)-dimensional nonlinear space. The method was used to estimate the impulse response parameters of ETSI BRAN B and Macchi channels. Note that based on the numerical simulation results, we have obtained good results, compared with the recursive projection (RP) algorithm. It was demonstrated that the proposed KRPI algorithm is capable of estimating the parameters of the finite impulse response system with a good level of accuracy, higher than that achieved by the RP algorithm whose performance deteriorates considerably. As far as channel impulse response identification is concerned, the proposed KRPI algorithm is effective and efficient in identifying the amplitude and phase of the channels (ETSI BRAN B and Macchi) with various SNR values.

The future work will focus on the stability and convergence of our solution in the context of non-line-of-sight outdoor channel identification for wireless sensor networks based on impulse ultra-wide band radio.

References

[1] S. Haykin, *Adaptive Filter Theory*, 4th ed. Prentice Hall, Delhi, 2002 (ISBN: 9780130901262).

[2] M.M. Sondhi, "The history of echo cancellation", *IEEE Signal Processing Magazine*, vol. 23, no. 5, pp. 95–102, 2006 (https://doi.org/10.1109/MSP.2006.1708416).

[3] A.H. Sayed, *Fundamentals of Adaptive Filtering*. New York: John Wiley & Sons, 824 p., 2003 (ISBN: 9780471461265).

[4] P.S.R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementations*, 3rd ed. New York: Springer, 495 p., 2008 (https://doi.org/10.1007/978-3-030-29057-3).

[5] R. Fateh, A. Darif, and S. Safi, "Kernel and linear adaptive methods for the BRAN channels identification", *AI2SD: International Conference on Advanced Intelligent Systems for Sustainable Development*, vol. 2, pp. 579–591, 2020 (https://doi.org/10.1007/978-3-030-90633-7).

[6] R. Fateh, A. Darif, and S. Safi, "Identification of the linear dynamic parts of wiener model using kernel and linear adaptive", *AI2SD: International Conference on Advanced Intelligent Systems for Sustainable Development*, vol. 2, pp. 387–400, 2020 (https://doi.org/10.1007/978-3-030-90633-7).

[7] E. Ferrara, "Fast implementations of LMS adaptive filters", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 474–475, 1980 (https://doi.org/10.1109/tassp.1980.1163432).

[8] M. Zidane and R. Dinis, "A new combination of adaptive channel estimation methods and TORC equalizer in MC-CDMA systems", *International Journal of Communication Systems*, vol. 33, no. 11, e4429, 2020 (https://doi.org/10.1002/dac.4429).

[9] M. Zidane, S. Safi, and M. Sabri, "Measured and estimated data of nonlinear BRAN channels using HOS in 4G wireless communications", *Data in Brief*, vol. 17, pp. 1136–1148, 2018 (https://doi.org/10.1016/j.dib.2018.02.005).

[10] R. Fateh, A. Darif, and S. Safi, "Performance Evaluation of MC-CDMA Systems with Single User Detection Technique using Kernel and Linear Adaptive Method", *Journal of Telecommunications and Information Technology*, no. 4, pp. 1–11 2021 (https://doi.org/10.26636/jtit.2021.151621).

[11] M. Zidane, S. Safi, and M. Sabri, "Compensation of fading channel using partial combining equalizer in MC-CDMA systems", *Journal of Telecommunications and Information Technology*, no. 1, pp. 5–11, 2017 [Online]. Available: https://www.il-pib.pl/czasopisma/JTIT/2017/1/5.pdf

[12] S. Safi, M. Frikel, A. Zeroual, and M. M'Saad, "Higher order cumulants for identification and equalization of multicarrier spreading spectrum systems", *Journal of Telecommunications and Information Technology*, no. 2, pp. 74–84, 2011 [Online]. Available: https://www.il-pib.pl/czasopisma/JTIT/2011/2/74.pdf

[13] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River (NJ): Prentice Hall PTR, 1999 (ISBN: 9780136566953).

[14] F. Ding, *System Identification – Performances Analysis for Identification Methods*. Beijing, China: Science Press, pp. 1–12, 2014 [in Chinese].

[15] X. Zhang, F. Ding, F.E. Alsaadi, and T. Hayat, "Recursive parameter identification of the dynamical models for bilinear state space systems", *Nonlinear Dynamics*, vol. 89, no. 4, pp. 2415–2429, 2017 (https://doi.org/10.1007/S11071-017-3594-Y).

[16] L. Xu, "Application of the Newton iteration algorithm to the parameter estimation for dynamical systems", *Journal of Computational and Applied Mathematics*, vol. 288, pp. 33–43, 2015 (https://doi.org/10.1016/j.cam.2015.03.057).

[17] Q. Song, "Recursive identification of systems with binary-valued out-puts and with ARMA noises", *Automatica*, vol. 93, pp. 106–113, 2018 (https://doi.org/10.1016/j.automatica.2018.03.059).

[18] J. Guo, X. Wang, W. Xue, and Y. Zhao, "System identification with binary-valued observations under data tampering attacks", *IEEE Transactions on Automatic Control*, vol. 66, no. 8, pp. 3825–3832, 2020 (https://doi.org/10.1109/TAC.2020.3029325).

[19] R. Fateh, A. Darif, and S. Safi, "Channel identification of non-linear systems with binary-valued output observations based on positive definite kernels", *E3S Web of Conferences*, vol. 297, 2021 (https://doi.org/10.1051/e3sconf/202129701020).

[20] L. Li, F. Wang, H. Zhang, and X. Ren, "A novel recursive learning estimation algorithm of Wiener systems with quantized observations", *ISA Transactions*, vol. 112, pp. 23–34, 2021 (https://doi.org/10.1016/j.isatra.2020.11.032).

[21] L. Zhang, Y. Zhao, and L. Guo, "Identification and adaptation with binary-valued observations under non-persistent excitation condition", *Automatica*, vol. 138, 2022 (https://doi.org/10.48550/arXiv.2107.03588).

[22] R. Fateh and A. Darif, "Mean square convergence of reproducing kernel for channel identification: Application to Bran D Channel impulse response", in *International Conference on Business Intelligence*, vol. 416, pp. 284–293, 2021 (https://doi.org/10.1007/978-3-030-76508-8_20).

[23] J.F. Zhang and G.G. Yin, "System identification using binary sensors", *IEEE Transactions on Automatic Control*, vol. 48, no. 11, pp. 1892–1907, 2003 (https://doi.org/10.1109/TAC.2003.819073).

[24] L.Y. Wang, G.G. Yin, and J.F. Zhang, "Joint identification of plant rational models and noise distribution functions using binary-valued observations", *Automatica*, vol. 42, no. 4, pp. 535–547, 2006 (https://doi.org/10.1016/j.automatica.2005.12.004).

[25] J. Guo, Y. Zhao, C.Y. Sun, and Y. Yu, "Recursive identification of FIR systems with binary-valued outputs and communication channels", *Automatica*, vol. 60, pp. 165–172, 2015 (https://doi.org/10.1016/j.automatica.2015.06.030).

[26] J. Guo and Y. Zhao, "Recursive projection algorithm on FIR system identification with binary-valued observations", *Automatica*, vol. 49, no. 11, pp. 3396–3401, 2013 (https://doi.org/10.1016/j.automatica.2013.08.011).

[27] M. Pouliquen, E. Pigeon, O. Gehan, A. Goudjil, and R. Auber, "Impulse response identification from input/output binary measurements", *Automatica*, vol. 123, art. no. 109307, 2021 (https://doi.org/10.1016/j.automatica.2020.109307).

[28] M. Pouliquen, E. Pigeon, O. Gehan, and A. Goudjil, "Identification using binary measurements for IIR systems", *IEEE Transactions*

on Automatic Control, vol. 65, no. 2, pp. 786–793, 2019 (https://doi.org/10.1109/TAC.2019.2921657).

[29] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004 (https://doi.org/10.1017/CBO9780511809682).

[30] W. Liu, J.C. Principe, and S. Haykin, *Kernel Adaptive Filtering: A Comprehensive Introduction*. John Wiley & Sons, 240 p., 2011 (ISBN: 9780470447536).

[31] R. Fateh, A. Darif and S. Safi, "Hyperbolic functions impact evaluation on channel identification based on recursive kernel algorithm", *2022 8th International Conference on Optimization and Applications (ICOA)*, Genoa, Italy, pp. 1–6, 2022 (https://doi.org/10.1109/ICOA55659.2022.9934118).

[32] J.W. Xu, A.R. Paiva, M. Park, and J.C. Principe, "A reproducing kernel Hilbert space framework for information-theoretic learning", *IEEE Transactions on Signal Processing*, vol. 56, no. 12, pp. 5891–5902, 2008 (https://doi.org/10.1109/TSP.2008.2005085).

[33] F. Girosi, M. Jones, and T. Poggio, "Regularization theory and neural networks architectures", *Neural Computation*, vol. 7, no. 2, pp. 219–269, 1995 (https://doi.org/10.1162/neco.1995.7.2.219).

[34] C.K.I. Williams and C.E. Rasmussen, *Gaussian Processes for Machine Learning*. Cambridge: MIT Press, 2005 (https://doi.org/10.7551/mitpress/3206.001.0001).

[35] C. Cortes, and V. Vapnik, "Support-vector networks", *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995 (https://doi.org/10.1007/bf00994018).

[36] W. Liu, J.C. Principe, "Kernel affine projection algorithms", *EURASIP Journal on Advances in Signal Processing*, no. 1, pp. 1–12, 2008 (https://doi.org/10.1155/2008/784292).

[37] B. Scholkopf, A. Smola, and K.R. Muller, "Kernel principal component analysis", in *International Conference on Artificial Neural Networks (ICANN)*, Lausanne, Switzerland, pp. 583–588, 1997 (https://doi.org/10.1007/BFb0020217).

[38] W. Liu, P. P. Pokharel, and J. C. Principe, "The kernel least-mean-square algorithm", *IEEE Transactions on Signal Processing*, vol. 56, no. 2, pp. 543–554, 2008 (https://doi.org/10.1109/tsp.2007.907881).

[39] Y. Engel, S. Mannor, and R. Meir, "The kernel recursive least-squares algorithm", *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2275–2285, 2004 (https://doi.org/10.1109/TSP.2004.830985).

[40] B. Chen, S. Zhao, P. Zhu, and J.C. Principe, "Quantized kernel recursive least squares algorithm", *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 9, pp. 1484–1491, 2013 (https://doi.org/10.1109/tnnls.2013.2258936).

[41] 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, no. 1, pp. 22–32, 2011 (https://doi.org/10.1109/tnnls.2011.2178446).

[42] W. Liu, I. Park, Y. Wang, and J.C. Principe, "Extended kernel recursive least squares algorithm", *IEEE Transactions on Signal Processing*, vol. 57, no. 10, pp. 3801–3814, 2009 (https://doi.org/10.1109/tsp.2009.2022007).

[43] B. Chen, J. Liang, N. Zheng, and J.C. Príncipe, "Kernel least mean square with adaptive kernel size", *Neurocomputing*, vol. 191, pp. 95–106, 2016 (https://doi.org/10.1016/j.neucom.2016.01.004).

[44] Z. Qin, B. Chen, and N. Zheng, "Random Fourier feature kernel recursive least squares", in *2017 International Joint Conference on Neural Networks (IJCNN)*, Anchorage, AK, USA, pp. 2881–2886, 2017 (https://doi.org/10.1109/IJCNN.2017.7966212).

[45] Q. Wu, Y. Li, Y.V. Zakharov, and W. Xue, "Quantized kernel Least lncosh algorithm", *Signal Processing*, vol. 189, art. no. 108255, 2021 (https://doi.org/10.1016/j.sigpro.2021.108255).

[46] R. Fateh, A. Darif, and S. Safi, "An extended version of the proportional adaptive algorithm based on kernel methods for chan-nel identification with binary measurements", *Journal of Telecommunications and Information Technology*, no. 3, pp. 47–58, 2022 (https://doi.org/10.26636/jtit.2022.161122).

[47] J. Guo and Y. Zhao, "Identification of the gain system with quantized observations and bounded persistent excitations", *Science China Information Sciences*, vol. 57, no. 1, pp. 1–15, 2014 (https://doi.org/10.1007/s11432-012-4761-x).

[48] J.D. Diao, J. Guo, C.Y. Sun, "Event-triggered identification of FIR systems with binary-valued output observations", *Automatica*, vol. 98, pp. 95–102, 2018 (https://doi.org/10.1016/j.automatica.2018.09.024).

[49] N. Aronszajn, "Theory of reproducing kernels", *Transactions of the American Mathematical Society*, vol. 68, no. 3, pp. 337–404, 1950 (https://doi.org/10.2307/1990404).

[50] C.J.C. Burges, "A tutorial on support vector machines for pattern recognition", *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, 1998 (https://doi.org/10.1023/a:1009715923555).

[51] ETSI. Broadband Radio Access Network (BRAN); Hiperlan type 2; Physical (PHY) layer, Technical Report, December 2001.

[52] ETSI. Broadband Radio Access Network (BRAN); Hiperlan type 2; Requirements and architectures for wireless broadband access, January, 1999.

[53] O. Macchi, C.A. Faria da Rocha, and J.M. Travassos-Romano, "Égalisation adaptative autodidacte par rétroprédiction et prédiction", *XIV Colloque GRETSI*, pp. 491–493, 1993 [in French].

———————————

Rachid Fateh, Ph.D.
https://orcid.org/0000-0002-0574-2105
E-mail: fateh.smi@gmail.com
Laboratory of Innovation in Mathematics, Applications and Information Technologies (LIMATI), Polydisciplinary Faculty, Sultan Moulay Slimane University, Beni Mellal, Morocco

Anouar Darif, Ph.D.
https://orcid.org/0000-0001-8026-9189
E-mail: anouar.darif@gmail.com
Laboratory of Innovation in Mathematics, Applications and Information Technologies (LIMATI), Polydisciplinary Faculty, Sultan Moulay Slimane University, Beni Mellal, Morocco

Ahmed Boumezzough, Ph.D.
https://orcid.org/0000-0002-0574-2105
E-mail: ahmed.boumezzough@usms.ma
Laboratory of Research in Physics and Engineering Sciences, Polydisciplinary Faculty, Sultan Moulay Slimane University, Beni Mellal, Morocco

Said Safi, Ph.D.
https://orcid.org/0000-0003-3390-9037
E-mail: safi.said@gmail.com
Laboratory of Innovation in Mathematics, Applications and Information Technologies (LIMATI), Polydisciplinary Faculty, Sultan Moulay Slimane University, Beni Mellal, Morocco

Miloud Frikel, Ph.D.
https://orcid.org/0000-0003-2178-1814
E-mail: miloud.frikel@ensicaen.fr
Laboratoire d'Automatique de Caen, UNICAEN, ENSICAEN, Normandie University, Caen, France