WIDELY LINEAR KERNEL-BASED ADAPTIVE FILTERS

P. Bouboulis[a], *S. Theodoridis*[a], *M. Mavroforakis*[b]

(a) Department of Informatics and Telecommunications, University of Athens Greece,
{bouboulis, stheodor}@di.uoa.gr.
(b) Computational Biomedicine Lab, Department of Computer Science, University of Houston, Texas,
mmavrof@uh.edu.

ABSTRACT

Widely linear estimation for complex-valued signal processing is growing in popularity, especially in the cases where the involved signals exhibit non-circular characteristics. In this paper, the extended Wirtinger's calculus in complex Reproducing Kernel Hilbert Spaces (RKHS), presented in [1], is adopted to derive complex kernel-based widely-linear estimation filters. Furthermore, we illuminate several important characteristics of widely linear filters, which, to our knowledge, haven't been considered before. Our results indicate that, in contrast to many cases where the gains from adopting widely linear estimation filters, instead of ordinary linear filters, are rudimentary, for the case of kernel-based widely linear filters significant performance improvements can be obtained.

1. INTRODUCTION

Complex-valued signals arise frequently in many signal processing applications. In contrast to the traditional splitting approach into the real and imaginary parts, the complex domain provides a convenient and elegant representation for these signals and also a natural way to preserve their characteristics as well as to perform transformations in an efficient way. Therefore, signal representations using complex algebra are often met in the literature. However, in many applications, one is often forced to make certain assumptions for the complex signal in order to simplify the algebra. One such assumption, which is commonly adopted, is the *circularity* of the signal. Circularity is intimately related to rotation in the geometric sense. A complex random variable, Z, is called circular, if for any angle ϕ both Z and $Ze^{i\phi}$ (i.e., the rotation of Z by angle ϕ) follow the same probability distribution [2, 3]. Naturally, this assumption limits the area of applications, since many practical signals exhibit non-circular characteristics. Thus, following the ideas originated by Picinbono in [4, 5], on-going research is focusing on the *widely linear* filters (or *augmented* filters) in the complex domain (see for example [2, 3, 6–13]). The main characteristic of such filters is that they exploit both the original signal as well as its conjugate analogue.

On the other hand, kernel-based processing is another topic that is gaining in popularity within the signal processing community [14–21], as it provides an efficient toolbox for treating nonlinear problems, transforming them into a space of higher dimensionality, possibly infinite, $\mathscr{H}$. However, all the popular kernel-based techniques were designed to process real data. Until recently, no kernel-based methodology for treating complex signals had been developed, in spite of their potential interest in a number of applications. Recently, in [1], a framework based on complex RKHS was presented to solve this problem. Its main contributions are: a) the development of a wide framework that allows real-valued kernel algorithms to be extended to treat complex data effectively, taking advantage of a technique called *complexification* of real RKHSs, b) the elevation from obscurity of the pure complex kernels (such as the complex Gaussian one) as a tool for kernel based adaptive processing of complex signals and c) the extension of *Wirtinger's Calculus* in complex RKHSs, as a means for an elegant and efficient computation of the gradients, which are involved in the derivation of adaptive learning algorithms.

In this paper, we adopt the main concepts of widely linear estimation, to develop kernel based widely linear adaptive filters using the framework presented in [1]. Moreover, we illuminate certain aspects of widely linear estimation from a new perspective that sheds light to why widely linear estimation is expected to perform better than complex linear estimation. We demonstrate that, in the context of pure complex kernels, adopting the widely linear filtering structure leads to a significantly improved performance. In contrast, combining the widely linear structure with kernels that result from complexification of real kernels, does not enhance performance, compared to that obtained with the ordinary kernel-based filters. This is because complexification implicitly adds a conjugate component to the model.

The paper is organized as follows. We start with a brief introduction to complex RKHSs in Section 2, before we review the framework of [1] in Section 3. The main notions of Wirtinger's calculus, which is mobilized to calculate the gradients of the respective cost functions, are also presented there. In Section 4, we describe the concept of widely linear estimation and show why this is better than complex linear estimation. Finally, widely linear kernel based adaptive filters are described in section 5. Experiments are provided in section 6. Throughout the paper, we will denote the set of all integers, real and complex numbers by $\mathbb{N}$, $\mathbb{R}$ and $\mathbb{C}$ respectively. Vector or matrix valued quantities appear in boldfaced symbols.

2. PRELIMINARIES

Let X be a nonempty set. Then a function $\kappa : X \times X \to \mathbb{F}$, which for all $N \in \mathbb{N}$ and all $x_1, \ldots, x_N \in X$ gives rise to a positive definite Gram matrix K, is called a *Positive Definite Kernel*. We can define a RKHS [22] as a Hilbert space $\mathscr{H}$ over a field $\mathbb{F}$ for which there exists a positive definite function $\kappa : X \times X \to \mathbb{F}$ with the following two important properties:

1. For every $x \in X$, $\kappa(\cdot, x)$ belongs to $\mathscr{H}$.
2. κ has the so called *reproducing property*, i.e.,

$$f(x) = \langle f, \kappa(\cdot, x)\rangle_{\mathscr{H}}, \text{ for all } f \in \mathscr{H}, \tag{1}$$

in particular $\kappa(x, y) = \langle \kappa(\cdot, y), \kappa(\cdot, x)\rangle_{\mathscr{H}}$.

The map $\Phi : X \to \mathscr{H} : \Phi(x) = \kappa(\cdot, x)$ is called the *feature map* of $\mathscr{H}$. Recall, that in the case of complex Hilbert spaces (i.e., $\mathbb{F} = \mathbb{C}$) the inner product is sesqui-linear (i.e., linear in one argument and antilinear in the other) and Hermitian. In the real case, the condition $\kappa(x, y) = \langle \kappa(\cdot, y), \kappa(\cdot, x)\rangle_{\mathscr{H}}$ may be replaced by $\kappa(x, y) = \langle \kappa(\cdot, x), \kappa(\cdot, y)\rangle_{\mathscr{H}}$. However, since in the complex case the inner product is Hermitian, the aforementioned condition is equivalent to $\kappa(x, y) = \left(\langle \kappa(\cdot, x), \kappa(\cdot, y)\rangle_{\mathscr{H}}\right)^*$.

Although, the underlying theory has been developed by the mathematicians for general complex reproducing kernels and their associated RKHSs, it is the real kernels that have been considered by the machine learning and signal processing communities. Some of the most widely used kernels in the literature are the *Gaussian Radial Basis Function (RBF)*, i.e., $\kappa_{\sigma,\mathbb{R}^d}(x, y) := \exp\left(-\sum_{i=1}^{d}(x_i - y_i)^2/\sigma^2\right)$, defined for $x, y \in \mathbb{R}^d$, where σ is a free positive parameter, and the *polynomial kernel*: $\kappa_d(x, y) := \left(1 + x^T y\right)^d$, for $d \in \mathbb{N}$. Many more can be found in [23, 24].

There are many complex reproducing kernels, that have been extensively studied by the mathematicians (see for example [25]). Here we focus our attention on the *complex Gaussian kernel*, which is defined as follows:

$$\kappa_{\sigma,\mathbb{C}^d}(z,w) := \exp\left(-\frac{\sum_{i=1}^{d}(z_i - w_i^*)^2}{\sigma^2}\right), \tag{2}$$

where $z,w \in \mathbb{C}^d$, z_i denotes the i-th component of the complex vector $z \in \mathbb{C}^d$ and exp is the extended exponential function in the complex domain. Its restriction, $\kappa_\sigma := \left(\kappa_{\sigma,\mathbb{C}^d}\right)_{|\mathbb{R}^d\times\mathbb{R}^d}$, is the well known *real Gaussian kernel*. An explicit description of the RKHSs of these kernels can be found in [26].

3. KERNEL PROCESSING IN COMPLEX RKHS

To generate kernel adaptive filtering algorithms on complex domains, according to [1], one can adopt two methodologies. A first straightforward approach is to use directly a complex RKHS, using one of the complex kernels and map the original data to the complex RKHS through the associated feature map $\Phi(z) = \kappa(\cdot,z)$. Another alternative technique is to use real kernels through a rationale that is called *complexification* of real RKHSs. This method has the advantage of allowing modeling in complex RKHSs, using popular well-established and well understood, from a performance point of view, real kernels (e.g., gaussian, polynomial, etc.).

Let $X \subseteq \mathbb{R}^\nu$. Define $X^2 \equiv X \times X \subseteq \mathbb{R}^{2\nu}$ and $\mathbb{X} = \{x+iy, x,y \in X\} \subseteq \mathbb{C}^\nu$ equipped with a complex product structure. Let $\mathscr{H}$ be a real RKHS associated with a real kernel κ defined on $X^2 \times X^2$ and let $\langle\cdot,\cdot\rangle_{\mathscr{H}}$ be its corresponding inner product. Then, every $f \in \mathscr{H}$ can be regarded as a function defined on either X^2 or $\mathbb{X}$, i.e., $f(z) = f(x+iy) = f(x,y)$. Next, we define $\mathscr{H}^2 = \mathscr{H} \times \mathscr{H}$. It is easy to verify that $\mathscr{H}^2$ is also a Hilbert Space with inner product $\langle f,g\rangle_{\mathscr{H}^2} = \langle f_1,g_1\rangle_{\mathscr{H}} + \langle f_2,g_2\rangle_{\mathscr{H}}$, for $f = (f_1,f_2)$, $g = (g_1,g_2)$. Our objective is to enrich $\mathscr{H}^2$ with a complex structure. To this end, we define the space $\mathbb{H} = \{f = f_1 + if_2;\ f_1,f_2 \in \mathscr{H}\}$ equipped with the complex inner product: $\langle f,g\rangle_{\mathbb{H}} = \langle f_1,g_1\rangle_{\mathscr{H}} + \langle f_2,g_2\rangle_{\mathscr{H}} + i(\langle f_2,g_1\rangle_{\mathscr{H}} - \langle f_1,g_2\rangle_{\mathscr{H}})$, for $f = f_1 + if_2$, $g = g_1 + ig_2$. Hence, $f,g : \mathbb{X} \subseteq \mathbb{C}^\nu \to \mathbb{C}$. It is not difficult to verify that $\mathbb{H}$ is a complex RKHS with kernel κ [25]. We call $\mathbb{H}$ the complexification of $\mathscr{H}$. It can readily be seen, that, although $\mathbb{H}$ is a complex RKHS, its respective kernel is real (i.e., its imaginary part is equal to zero).

To complete the presentation of the required complexification framework, we need a technique to implicitly map the data samples from the complex input space to the complexified RKHS $\mathbb{H}$. This can be done using the simple rule: $\hat{\Phi}(z) = \hat{\Phi}(x+iy) = \hat{\Phi}(x,y) = \Phi(x,y) + i\Phi(x,y)$, where Φ is the feature map of the real reproducing kernel κ, i.e., $\Phi(x,y) = \kappa(\cdot,(x,y))$. It must be emphasized, that $\hat{\Phi}$ is not the feature map associated with the complex RKHS $\mathbb{H}$. Furthermore, the intersting point is that in spite of the fact that the space is a complex one, the "generating" kernel is a real one. Therefore, the algorithms derived using this approach cannot be reproduced, if one blindly applies the kernel trick and replaces complex inner products in a linear algorithm, that works directly on the data in the input space, in order to kernelize it [1].

3.1 Wirtinger's Calculus in complex RKHS

Wirtinger's calculus [27] was brought into light recently [2–4, 7, 28, 29], as a means to compute, in an efficient and elegant way, gradients of real valued cost functions that are defined on complex domains ($\mathbb{C}^\nu$). It is based on simple rules and principles, which bear a great resemblance to the rules of the standard complex derivative, and it greatly simplifies the calculations. The difficulty with real valued cost functions is that they do not obey the Cauchy-Riemann conditions and are not diffirentiable in the complex domain. The alternative to Wirtinger's calculus would be to consider the complex variables as pairs of two real ones and employ the common real partial derivatives. However, this approach, usually, is more time consuming and leads to more cumbersome expressions.

In [1], the notions of Wirtinger's calculus was extended to general complex Hilbert spaces, providing the tool to compute the gradients that are needed to develop kernel-based algorithms for treating complex data. This extension uses mainly the notion of the *Fréchet differentiability*, which generalizes differentiability to general Hilbert spaces. However, due to lack of space, in this section we give only the definitions and the main results that will be used to derive the widely linear kernel-based estimation filters. The interested reader may find more on the subject in [1, 30].

Consider the function $T : A \subseteq \mathbb{H} \to \mathbb{C}$, $T(f) = T(u_f + iv_f) = T_r(u_f,v_f) + iT_i(u_f,v_f)$, defined on an open subset A of $\mathbb{H}$, where $u_f, v_f \in \mathscr{H}$ and T_r, T_i are real valued functions defined on $\mathscr{H}^2$. Any such function, T, may be regarded as defined either on a subset of $\mathbb{H}$, or on a subset of $\mathscr{H}^2$. Moreover, T may be regarded either as a complex valued function, or as a vector valued function, which takes values in $\mathbb{R}^2$. Therefore, we may equivalently write: $T(f) = T(u_f + iv_f) = T_r(u_f,v_f) + iT_i(u_f,v_f)$, or $T(f) = \left(T_r(u_f,v_f), T_i(u_f,v_f)\right)^T$.

We define the *Fréchet Wirtinger's gradient* (or *W-gradient* for short) and the *Fréchet conjugate Wirtinger's gradient* (or *CW-gradient* for short) of T at c as follows:

$$\begin{aligned}\nabla_f T(c) =& \frac{1}{2}\left(\nabla_1 T(c) - i\nabla_2 T(c)\right) = \frac{1}{2}\left(\nabla_u T_r(c) + \nabla_v T_i(c)\right) \\ &+ \frac{i}{2}\left(\nabla_u T_i(c) - \nabla_v T_r(c)\right), \end{aligned} \tag{3}$$

$$\begin{aligned}\nabla_{f^*} T(c) =& \frac{1}{2}\left(\nabla_1 T(c) + i\nabla_2 T(c)\right) = \frac{1}{2}\left(\nabla_u T_r(c) - \nabla_v T_i(c)\right) \\ &+ \frac{i}{2}\left(\nabla_u T_i(c) + \nabla_v T_r(c)\right). \end{aligned} \tag{4}$$

The rules of the generalized calculus can be found in [1, 30]. Here we focus our interest to the following two simple properties:

1. If $T(f) = \langle f,w\rangle_{\mathbb{H}}$, then $\nabla_f T(c) = w^*$, $\nabla_{f^*} T(c) = 0$, for every c.
2. If $T(f) = \langle f^*,w\rangle_{\mathbb{H}}$, then $\nabla_f T(c) = 0$, $\nabla_{f^*} T(c) = w^*$, for every c.

In [1], the aforementioned toolbox was employed in the context of the complex Least Mean Square (LMS) algorithm and two realizations of the complex kernel LMS algorithm were developed. The first one, which will be denoted here as NCKLMS1 adopts the complexification procedure and the second one, which will be denoted as NCKLMS2 uses the complex gaussian kernel.

4. WIDELY LINEAR ESTIMATION FILTERS

In this paper, our attention is focussed on the *widely linear* estimation filters, or *augmented* filters, as they are also known. These filters take into account both the original values of the signal data as well as their conjugates. For example, in a typical LMS task, we estimate the output as $\hat{d}(n) = w^H z(n)$ and the step update as $w(n) = w(n-1) + \mu e^*(n) z(n)$. In this case, $\hat{d}(n)$ is provided as the output of a linear estimation filter. However, the linearity property is taken with respect to the field of complex numbers. Picinbono and Chevalier, in [4], proposed an alternative approach. They estimated the filter's output as $\tilde{d}(n) = w^H z + v^H z^*$ and showed that it provides better results in terms of the mean square error. This, of course, is expected since $\tilde{d}(n)$ provides a more rich representation than $\hat{d}(n)$. On the other hand, $\tilde{d}(n)$ is no longer linear over the field $\mathbb{C}$. It is linear, however, over the real numbers $\mathbb{R}$. To emphasize this difference, in the relative literature $\hat{d}(n)$ is often called $\mathbb{C}$-linear, while $\tilde{d}(n)$ is called $\mathbb{R}-linear$.

In [4, 6, 31], it is shown that the widely linear estimation filter is able to capture the second order statistical characteristics of the signal which are essential, especially for non-circular sources. Although in many relative works this is highlighted as the main reason for adopting widely linear techniques, in this paper, we will highlight a different perspective.

Our starting point will be the definition of linearity, in its strict mathematical sense. To this end, let us first clarify, that complex processing is equivalent with processing two real signals in the respective Euclidean (Hilbert) spaces. The advantage of using complex algebra is that the algorithm and/or the solution may be described in a more compact form. Moreover, the complex algebra allows for a more intuitive understanding of the problem, as many geometric transformations can be easily described using complex algebra in an elegant way. Finally, the application of Wirtinger's calculus greatly simplifies the calculations needed for the gradients of real valued cost functions.

Having this in mind, we now turn our attention to a typical complex LMS task. Let $z(n) \in \mathbb{C}^\nu$ and $d(n) \in \mathbb{C}$ be the input and the output of the original filter. We estimate the output of the filter using a $\mathbb{C}$-linear response $\hat{d}(n) = w^H z(n)$. The typical complex LMS task aims to compute $w \in \mathbb{C}^\nu$, such that the error $|d(n) - \hat{d}(n)|^2$ is minimized. If we set $w = w_r + iw_i$ and $z = x + iy$, we take that

$$\hat{d}(n) = w_r^T x + w_i^T y + i(w_r^T y - w_i^T x). \tag{5}$$

However, the real essence behind a complex filter operation is the following: Given two real vectors, $x(n)$ and $y(n)$, compute linear filters in order to estimate two new real processes, $d_r(n)$ and $d_i(n)$, in an optimal way, that jointly cares for both $d_r(n)$ and $d_i(n)$. Let us express the problem in its multichannel formulation, using real variables only, i.e.,

$$\begin{pmatrix} \tilde{d}_r(n) \\ \tilde{d}_i(n) \end{pmatrix} = \begin{pmatrix} u_{1,1}^T & u_{1,2}^T \\ u_{2,1}^T & u_{2,2}^T \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} \equiv U \cdot \begin{pmatrix} x \\ y \end{pmatrix}$$

From a mathematical point of view, this is the definition of a linear operator from $\mathbb{R}^{2\nu} \to \mathbb{R}^2$. The elements of U are computed such that both $|d_r(n) - \tilde{d}_r(n)|^2$ and $|d_i(n) - \tilde{d}_i(n)|^2$ are jointly minimized. This leads to the so called Dual Real Channel (DRC) formulation. Thus, we take the relations $\tilde{d}_r(n) = u_{1,1}^T x + u_{1,2}^T y$ and $\tilde{d}_i(n) = u_{2,1}^T x + u_{2,2}^T y$. In complex notation, if we consider that $\tilde{d}(n) = \tilde{d}_r(n) + i\tilde{d}_i(n)$, we obtain the expression

$$\tilde{d}(n) = u_{1,1}^T x + u_{1,2}^T y + i\left(u_{2,1}^T x + u_{2,2}^T y\right). \tag{6}$$

It is easy to see that the DRC approach expressed by relation (6) adopts a richer representation than that of the traditional LMS in (5). Moreover, it takes a few lines of elementary algebra to derive that an equivalent expression of (6) is the widely linear estimation filter. This will be our kick off point to define the task in a general Hilbert space. In general, we can prove the following[1]:

Proposition 1. *Consider a real[2] Hilbert space $\mathscr{H}$ and let the real Hilbert space $\mathscr{H}^2$ and the complex Hilbert space $\mathbb{H}$ be defined as in section 3. Then any linear function $T : \mathscr{H}^2 \to \mathbb{R}^2$ can be expressed in complex notation as*

$$T(x,y) = T(x+iy) = T(z) = \langle z, w\rangle_{\mathbb{H}} + \langle z^*, v\rangle_{\mathbb{H}}, \tag{7}$$

for some $w, v \in \mathbb{H}$, where $\langle \cdot,\cdot\rangle_{\mathbb{H}}$ is the respective inner product of $\mathbb{H}$.

Remark 1. *In view of proposition 1, one understands that the original formulation of the complex LMS was unorthodox, as it excludes a large class of linear functions from being considered in the estimation process. It is evident, that the linearity with respect to the field of complex numbers is restricted, compared to the linearity that underlies the DRC approach, which is more natural. Thus the correct complex linear estimation is $T(z) = \langle z, w\rangle_{\mathbb{H}} + \langle z^*, v\rangle_{\mathbb{H}}$ rather than $T(z) = \langle z, w\rangle_{\mathbb{H}}$.*

[1] The proof is omitted due to lack of space. It will be presented elsewhere.

[2] By the term real (complex) Hilbert space, we mean a Hilbert space over the field of real numbers $\mathbb{R}$ (complex numbers $\mathbb{C}$).

5. WIDELY LINEAR ESTIMATION IN COMPLEX RKHS

In this section, we will develop several realizations of the *Augmented Complex Kernel LMS* (ACKLMS) algorithm using either pure complex kernels, or real kernels under the complexification trick. We show that ACKLMS offers significant improvements versus complex kernel LMS (CKLMS), when the complex gaussian kernel is employed. On the other hand, as we will see, the CKLMS that is developed around the complexification trick, implicitly includes the augmented estimation, as it is directly associated with the DRC approach. Therefore, its augmented version degenerates to the standard CKLMS.

Consider the sequence of examples $(z(1), d(1))$, $(z(2), d(2))$, $\dots$, $(z(N), d(N))$, where $d(n) \in \mathbb{C}$, $z(n) \in \mathbb{C}^\nu$, $z(n) = x(n) + iy(n)$, $x(n), y(n) \in \mathbb{R}^\nu$, for $n = 1, \dots, N$. Consider, also, a real RKHS $\mathscr{H}$ and let the real Hilbert space $\mathscr{H}^2$ and the complex Hilbert space $\mathbb{H}$ be defined as in section 3. The straightforward method that one may apply in order to derive a widely linear estimation filter in $\mathbb{H}$, is:

$$\tilde{d}(n) = \langle \Phi(z(n)), w(n-1)\rangle_{\mathbb{H}} + \langle \Phi^*(z(n)), v(n-1)\rangle_{\mathbb{H}}, \tag{8}$$

where Φ is an appropriate function that maps the input data to the feature space $\mathbb{H}$. This is equivalent with transforming the data to a complex RKHS and applying a widely linear complex LMS to the transformed data. The objective of the ACKLMS is to estimate, w and v, so that to minimize $E[\mathscr{L}_n(w)]$, where $\mathscr{L}_n(w) = |e(n)|^2 = \left|d(n) - \tilde{d}(n)\right|^2$, at each time instance n.

Applying the rules of Wirtinger's calculus in complex RKHS, we can easily deduce that $\frac{\partial \mathscr{L}(n)}{\partial w^*} = -\Phi(z(n)) \cdot e^*(n)$, $\frac{\partial \mathscr{L}(n)}{\partial v^*} = -\Phi^*(z(n)) \cdot e^*(n)$. Thus, the step updates of the ACKLMS are

$$\begin{aligned} w(n) &= w(n-1) + \mu\Phi(z(n))e^*(n), \\ v(n) &= v(n-1) + \mu\Phi^*(z(n))e^*(n). \end{aligned}$$

Assuming that $w(0) = v(0) = 0$, the repeated application of the weight-update equations gives:

$$w(n-1) = \mu \sum_{k=0}^{n-1} \Phi(z(n))e^*(n), \tag{9}$$

$$v(n-1) = \mu \sum_{k=0}^{n-1} \Phi^*(z(n))e^*(n). \tag{10}$$

Combining (8), (9) and (10) leads to

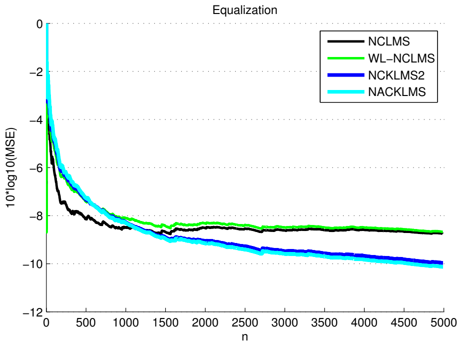
$$\tilde{d}(n) = \mu \sum_{k=1}^{n-1} e(k) \langle \Phi(z(n)), \Phi(z(k))\rangle_{\mathbb{H}} + \mu \sum_{k=1}^{n-1} e(k) \langle \Phi^*(z(n)), \Phi^*(z(k))\rangle_{\mathbb{H}}. \tag{11}$$

5.1 Complexified ACKLMS

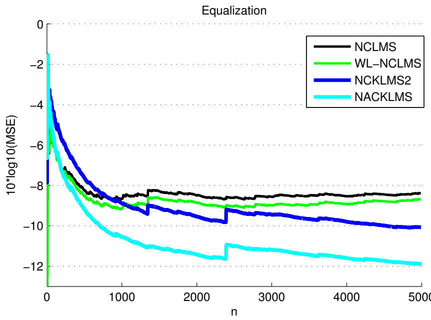
Recall that in the complexification trick, the associated function that maps the data to $\mathbb{H}$ is given by $\hat{\Phi}(z) = \Phi(x,y) + i\Phi(x,y) = \kappa_{\mathbb{R}}(\cdot,(x,y)) + i\kappa_{\mathbb{R}}(\cdot,(x,y))$, where Φ is the feature map of $\mathscr{H}$ and $\kappa_{\mathbb{R}}$ its respective real kernel. Under this condition, the filter output at iteration n takes the form

$$\tilde{d}(n) = 4\mu \sum_{k=1}^{n-1} e(k) \cdot \kappa_{\mathbb{R}}\left((x(k),y(k)),(x(n),y(n))\right). \tag{12}$$

This is exactly the same solution as in the complexified CKLMS (except a rescaling). Thus, we deduce that the standard complexified CKLMS presented in [1] and the augmented CKLMS are identical.



(a)



(b)

Figure 1: Learning curves for CKLMS2 ($\mu = 1/4$), NACKLMS, ($\mu = 1/4$), CLMS ($\mu = 1/16$) and widely linear CLMS ($\mu = 1/16$) (filter length $L = 5$, delay $D = 2$) for the soft nonlinear channel equalization problem, for (a) the circular input case, (b) the non-circular input case ($\rho = 0.1$).

5.2 ACKLMS with pure complex kernels

For the case of a pure complex kernel $\kappa_{\mathbb{C}}$, the filter output becomes

$$\tilde{d}(n) = \mu \sum_{k=1}^{n-1} e(k)\kappa_{\mathbb{C}}(z(k), z(n)) + \mu \sum_{k=1}^{n-1} e(k)\kappa^*_{\mathbb{C}}(z(k), z(n)), \quad (13)$$

as $\langle \Phi(z), \Phi(c) \rangle_{\mathbb{H}} = \kappa_{\mathbb{C}}(z, c)$, for any $z, c \in \mathbb{C}^\nu$. In this case, it is evident that the ACKLMS will give different solution to CKLMS, as it exploits a richer representation.

6. EXPERIMENTS

A normalized version of the Augmented CKLMS algorithm (denoted here as NACKLMS) was developed. Its performance has been tested in the context of a nonlinear channel equalization task. As in [1], two nonlinear channels have been considered. The first channel (labeled as *soft nonlinear channel* in the figures) consists of a linear filter: $t(n) = (-0.9 + 0.8i) \cdot s(n) + (0.6 - 0.7i) \cdot s(n-1)$ and a memoryless nonlinearity $q(n) = t(n) + (0.1 + 0.15i) \cdot t^2(n) + (0.06 + 0.05i) \cdot t^3(n)$. The second one (labeled as *strong nonlinear channel* in the figures) comprises the same linear filter and the nonlinearity: $q(n) = t(n) + (0.2 + 0.25i) \cdot t^2(n) + (0.12 + 0.09i) \cdot t^3(n)$. These are standard models, that have been extensively used in the literature for such tasks [15]. At the receiver end of the channels, the signal is corrupted by white Gaussian noise and then observed as $r(n)$. The level of the noise was set to 16dB. The input signal that was fed to the channels had the form $s(n) = 0.70\left(\sqrt{1-\rho^2}X(n) + i\rho Y(n)\right)$, where $X(n)$ and $Y(n)$ are gaussian random variables. This input is circular for $\rho = \sqrt{2}/2$ and highly non-circular if ρ approaches 0 or 1 [2].

The aim of a channel equalization task is to construct an inverse filter, which acts on the output $r(n)$ and reproduces the original input signal as closely as possible. To this end, we apply the NACKLMS algorithm to the set of samples $((r(n+D), r(n+D-1), \ldots, r(n+D-L+1)), s(n))$, where $L > 0$ is the filter length and D the equalization time delay, which is present to, almost, any equalization set up.

Experiments were conducted on a set of 5000 samples of the input signal considering both the circular and the non-circular cases. The results are compared with the NCLMS and the NACLMS (i.e., augmented NCLMS or widely linear NCLMS as it is also known) algorithms and with two adaptive nonlinear algorithms: a) the Complex non-linear Gradient descent (CNGD) algorithm [3] and a Multi Layer Perceptron (MLP) with 50 nodes in the hidden layer (proposed in [2]). In both cases, the complex tanh activation function was employed. For the case of the MLP, the design was also tuned so that the best possible results were obtained. Time delay D was set for optimality. Figure 1, shows the learning curves of the normalized CKLMS2 (NCKLMS2) and NACKLMS using the complex Gaussian kernel $\kappa_{\sigma,\mathbb{C}^d}(z, w) := \exp\left(-\frac{\sum_{i=1}^{d}(z_i - w_i^*)^2}{\sigma^2}\right)$ (with $\sigma = 5$), together with those obtained from the NCLMS and the NACLMS algorithms. We observe that the performance of NACKLMS and NCKLMS1 is similar, with the latter one leading to a smaller MSE value for that particular problem. Figure 2 shows the learning curves of NCKLMS2 and NACKLMS versus the CNGD and the L-50-1 MLP for the hard non-linear channel.

The novelty criterion (see [1], [14]) was used for the sparsification of the NCKLMS2 and NACKLMS with $\delta_1 = 0.1$ and $\delta_2 = 0.2$. In both examples, NACKLMS considerably outperforms the linear, widely linear (i.e., NCLMS and NACLMS) and nonlinear (CNGD and MLP) algorithms (see figures 1, 2). The NACKLMS also exhibits improved performance compared to the NCKLMS2 for non-circular input sources. Moreover, observe that while the gains of the ACLMS over CLMS are rather rudimentary (smaller than 0.2 dB), the gains of NACKLMS over NCKLMS2 are significant (approximately 2dB). For circular signals, the two models (NCKLMS2 and NACKLMS) lead to almost identical results, as expected [4, 5].

REFERENCES

[1] P. Bouboulis and S. Theodoridis, "Extension of Wirtinger's calculus to reproducing Kernel Hilbert Spaces and the complex kernel LMS," *IEEE Transactions on Signal Processing*, (to appear in 2011).

[2] T. Adali and H. Li, *Adaptive signal processing: next generation solutions*. Wiley, NJ, 2010, ch. Complex-valued Adaptive Signal Processing, pp. 1–74.

[3] D. Mandic and V. Goh, *Complex valued nonlinear adaptive filters*. Wiley, 2009.

[4] B. Picinbono and P. Chevalier, "Widely linear estimation with complex data," *IEEE Transactions on Signal Processing*, vol. 43, no. 8, pp. 2030–2033, 1995.

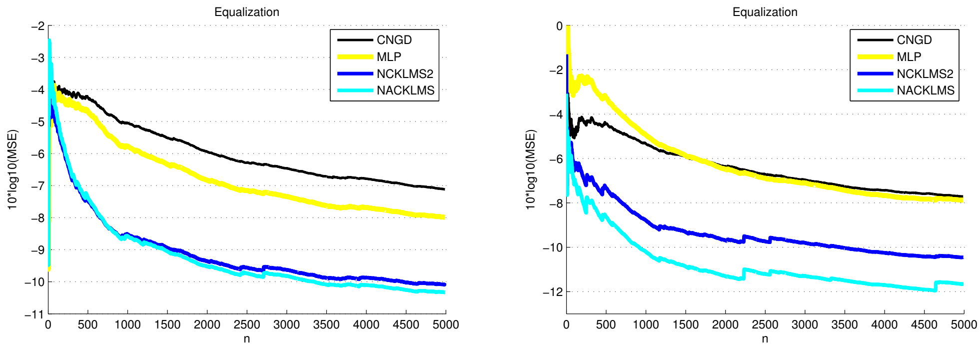
[5] B. Picinbono, "On circularity," *IEEE Transactions on Signal Processing*, vol. 42, no. 12, pp. 3473–3482, 1994.

[6] P. J. Schreier and L. L. Scharf, "Second-order analysis of improper complex random vectors and processes," *IEEE Trans. Signal Process.*, vol. 51, pp. 714–725, 2003.

[7] M. Novey and T. Adali, "On extending the complex ICA algorithm to noncircular sources," *IEEE Transanctions on Signal Processing*, vol. 56, no. 5, pp. 2148–2154, 2008.

[8] A. Aghaei, K. Plataniotis, and S. Pasupathy, "Widely linear MMSE receivers for linear dispersion space-time block-codes," *IEEE Transactions on Wireless Communications*, vol. 9, no. 1, pp. 8 – 13, 2010.

[9] A. Cacciapuoti, G. Gelli, L. Paura, and F. Verde, "Widely linear versus linear blind multiuser detection with subspace-based channel estimation: finite sample-size effects," *IEEE*



(a) (b)

Figure 2: Learning curves for CKLMS2 ($\mu = 1/4$), NACKLMS, ($\mu = 1/4$), MLP ($\mu = 1/16$) and CNGD ($\mu = 0.0005$) (filter length $L = 5$, delay $D = 2$) for the hard nonlinear channel equalization problem, for (a) the circular input case, (b) the non-circular input case ($\rho = 0.1$).

Transactions on Signal Processing, vol. 57, no. 4, pp. 1426 – 1443, 2008.

[10] K. Kuchi and V. Prabhu, "Performance evaluation for widely linear demodulation of PAM/QAM signals in the presence of Rayleigh fading and co-channel interference," *IEEE Transactions on Communications*, vol. 57, no. 1, pp. 183 – 193, 2009.

[11] J. Navarro-Moreno, J. Moreno-Kayser, R. Fernandez-Alcala, and J. Ruiz-Molina, "Widely linear estimation algorithms for second-order stationary signals," *IEEE Transactions on Signal Processing*, vol. 57, no. 12, pp. 4930 – 4935, 2009.

[12] F. Sterle, "Widely linear MMSE transceivers for MIMO channels," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 4258 – 4270, 2007.

[13] K. C. Pun and T. Nguyen, "Widely linear filter bank equalizer for real STBC," *IEEE Transactions on Signal Processing*, vol. 56, no. 9, pp. 4544 – 4548, 2008.

[14] W. Liu, J. C. Principe, and S. Haykin, *Kernel Adaptive Filtering*. Wiley, 2010.

[15] W. Liu, P. Pokharel, and J. C. Principe, "The kernel least-mean-square algorithm," *IEEE Trans. Signal Process.*, vol. 56, no. 2, pp. 543–554, 2008.

[16] J. Xu, A. Paiva, I. Park, and J. Principe, "A reproducing kernel Hilbert space framework for information theoretic learning," *IEEE Transactions on Signal Processing*, vol. 56, no. 12, pp. 5891–5902, 2008.

[17] J. Kivinen, A. Smola, and R. C. Williamson, "Online learning with kernels," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2165–2176, 2004.

[18] Y. Engel, S. Mannor, and R. Meir, "The kernel recursive least-squares algorithm," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2275–2285, 2004.

[19] K. Slavakis, S. Theodoridis, and I. Yamada, "On line classification using kernels and projection based adaptive algorithm," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 2781–2797, 2008.

[20] ——, "Adaptive constrained learning in reproducing kernel Hilbert spaces: the robust beamforming case," *IEEE Transactions on Signal Processing*, vol. 57, no. 12, pp. 4744–4764, 2009.

[21] P. Bouboulis, K. Slavakis, and S. Theodoridis, "Adaptive kernel-based image denoising employing semi-parametric regularization," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1465–1479, 2010.

[22] N. Aronszajn, "Theory of reproducing kernels," *Transactions of the American Mathematical Society*, vol. 68, pp. 337–404, 1950.

[23] B. Scholkopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, 2002.

[24] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 4^{th} ed. Academic Press, Nov. 2008.

[25] V. I. Paulsen, "An Introduction to the theory of Reproducing Kernel Hilbert Spaces," September 2009.

[26] I. Steinwart, D. Hush, and C. Scovel, "An explicit description of the Reproducing Kernel Hilbert spaces of Gaussian RBF kernels," *IEEE Trans. Info. Theory*, vol. 52, no. 10, pp. 4635–4643, 2006.

[27] W. Wirtinger, "Zur formalen theorie der functionen von mehr complexen veranderlichen," *Mathematische Annalen*, vol. 97, pp. 357–375, 1927.

[28] H. Li, "Complex-valued adaptive signal processing using Wirtinger calculus and its application to Independent Component Analysis," Ph.D. dissertation, University of Maryland Baltimore County, 2008.

[29] T. Adali, H. Li, M. Novey, and J. F. Cardoso, "Complex ICA using nonlinear functions," *IEEE Trans. Signal Process.*, vol. 56, no. 9, pp. 4536–4544, 2008.

[30] P. Bouboulis, "Wirtingers calculus in general hilbert spaces," *Tech Report, University of Athens*, 2010. [Online]. Available: http://arxiv.org/abs/1005.5170

[31] B. Picinbono and P. Bondon, "Second order statistics of complex signals," *IEEE Trans. Signal Process.*, vol. 45, no. 2, pp. 411–420, 1997.