Extension of Wirtinger's Calculus to Reproducing Kernel Hilbert Spaces and the Complex Kernel LMS

Pantelis Bouboulis, *Member, IEEE,* and Sergios Theodoridis, *Fellow, IEEE*

Abstract—Over the last decade, kernel methods for nonlinear processing have successfully been used in the machine learning community. The primary mathematical tool employed in these methods is the notion of the Reproducing Kernel Hilbert Space. However, so far, the emphasis has been on batch techniques. It is only recently, that online techniques have been considered in the context of adaptive signal processing tasks. Moreover, these efforts have only been focussed on real valued data sequences. To the best of our knowledge, no adaptive kernel-based strategy has been developed, so far, for complex valued signals. Furthermore, although the real reproducing kernels are used in an increasing number of machine learning problems, complex kernels have not, yet, been used, in spite of their potential interest in applications that deal with complex signals, with Communications being a typical example. In this paper, we present a general framework to attack the problem of adaptive filtering of complex signals, using either real reproducing kernels, taking advantage of a technique called *complexification* of real RKHSs, or complex reproducing kernels, highlighting the use of the complex gaussian kernel.

In order to derive gradients of operators that need to be defined on the associated complex RKHSs, we employ the powerful tool of Wirtinger's Calculus, which has recently attracted attention in the signal processing community. Wirtinger's calculus simplifies computations and offers an elegant tool for treating complex signals. To this end, in this paper, the notion of Wirtinger's calculus is extended, for the first time, to include complex RKHSs and use it to derive several realizations of the Complex Kernel Least-Mean-Square (CKLMS) algorithm. Experiments verify that the CKLMS offers significant performance improvements over several linear and nonlinear algorithms, when dealing with nonlinearities.

I. INTRODUCTION

Processing in Reproducing Kernel Hilbert Spaces (RKHSs), in the context of online learning, is gaining in popularity within the Machine Learning and Signal Processing communities [1]–[6]. The main advantage of mobilizing the tool of RKHSs is that the original nonlinear task is "transformed" into a linear one, which can be solved by employing an easier "algebra". Moreover, different types of nonlinearities can be treated in a unifying way, with no effect on the mathematical derivation of the algorithms, except at the final implementation stage. The main concepts of this

P. Bouboulis is with the Department of Informatics and Telecommunications, University of Athens, Greece, e-mail: bouboulis@di.uoa.gr.

S. Theodoridis is with the Department of Informatics and Telecommunications, University of Athens, Greece, and the Research Academic Computer Technology Institute, Patra, Greece. e-mail: stheodor@di.uoa.gr.

procedure can be summarized in the following two steps: 1) Map the finite dimensionality input data from the input space F (usually $F \subset \mathbb{R}^\nu$) into a higher dimensionality (possibly infinite) RKHS $\mathcal{H}$ and 2) Perform a linear processing (e.g., adaptive filtering) on the mapped data in $\mathcal{H}$. The procedure is equivalent with a non-linear processing (non-linear filtering) in F.

An alternative way of describing this process is through the popular *kernel trick* [7], [8]: Given an algorithm, which can be formulated in terms of dot products, one can construct an alternative algorithm by replacing each one of the dot products with a positive definite kernel κ. The specific choice of kernel implicitly defines a RKHS with an appropriate inner product. Furthermore, the choice of kernel also defines the type of nonlinearity that underlies the model to be used. The main representatives of this class of algorithms are the celebrated *support vector machines* (SVMs), which have dominated the research in machine learning over the last decade [9]. Besides SVMs and the more recent applications in adaptive filtering, there is a plethora of other scientific domains that have gained from adopting kernel methods (e.g., image processing and denoising [10], [11], principal component analysis [12], clustering [13], e.t.c.).

In classification tasks (which have been the dominant applications of kernel methods) the use of complex reproducing kernels is meaningless, since no arrangement can be derived in complex domains and the necessary separating hypersurfaces cannot be defined. Consequently, all known kernel based applications, as they emerged from the specific background, use real-valued kernels and they are able to deal with real valued data sequences only. To our knowledge, no kernel-based strategy has been developed, so far, that is able to effectively deal with complex valued signals.

In this paper, we present a general framework to address the problem of adaptive filtering of complex signals, using either real reproducing kernels, taking advantage of a technique called *complexification* of real RKHSs, or complex reproducing kernels, highlighting mostly the use of the complex gaussian kernel. Although the real gaussian RBF kernel has become quite popular and it has been used in many applications, the complex gaussian RBF kernel, while known to the mathematicians (especially those working on Reproducing Kernel Hilbert Spaces or Functional Analysis), it has rather remained in obscurity in the Machine Learning and Signal Processing communities. Even though the presented framework has a broad range and may be applied to generalize

a wide variety of kernel methods to the complex domain, this work focuses on the recently developed Kernel LMS (KLMS) [1], [14].

To compute the gradients of cost functions that are defined on the complex RKHSs, the principles of Wirtinger's calculus are employed. Wirtinger's calculus [15] has recently attracted attention in the signal processing community, mainly in the context of complex adaptive filtering [16]–[23], as a means of computing, in an elegant way, gradients of real valued cost functions defined on complex domains ($\mathbb{C}^{\nu}$). To this end, the main ideas and theorems of Wirtinger's calculus are generalized to general complex Hilbert spaces for the first time.

To summarize, the main contributions of this paper 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 *complex-ification* of real RKHSs, b) to elevate from obscurity the complex Gaussian kernel as a tool for kernel based adaptive processing of complex signals, 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, and d) the development of several realizations of the Complex Kernel LMS (CKLMS) algorithm, by exploiting the extension of Wirtinger's calculus and the generated complex RKHSs.

The paper is organized as follows. We start with an introduction to RKHSs in Section II, which includes real and complex kernels, before we briefly review the KLMS algorithm in Section III. In Section IV, we describe the complexification procedure of a real RKHS, that provides a framework to develop complex kernel methods, based on popular real valued reproducing kernels (e.g., gaussian, polynomial, e.t.c.). A brief introduction on Wirtinger's Calculus in finite dimensional spaces can be found in Section V. The main notions of the extended Wirtinger's Calculus on general Hilbert spaces are summarized in Section VI and the CKLMS is developed thereafter in Section VII. Finally, experimental results and conclusions are provided in Sections VIII and IX. 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.

II. REPRODUCING KERNEL HILBERT SPACES

In this section, we briefly describe the theory of Reproducing Kernel Hilbert Spaces. Since we are interested on both real and complex kernels, we recall the basic facts on RKHS associated with a general field $\mathbb{F}$, which can be either $\mathbb{R}$ or $\mathbb{C}$. However, we highlight the basic differences between the two cases. The material presented here may be found with more details in [24] and [25].

A. Basic Definitions

Given a function $\kappa : X \times X \to \mathbb{F}$ and $x_1, \ldots, x_N \in X$, the matrix[1] $K = (K_{i,j})^N$ with elements $K_{i,j} = \kappa(x_i, x_j)$, for

[1]The term $(K_{i,j})^N$ denotes a square $N \times N$ matrix.

$i, j = 1, \ldots, N$, is called the *Gram matrix* (or *kernel matrix*) of κ with respect to $x_1, \ldots, x_N$. A Hermitian matrix $K = (K_{i,j})^N$ satisfying

$$c^H \cdot K \cdot c = \sum_{i=1, j=1}^{N,N} c_i^* c_j K_{i,j} \geq 0,$$

for all $c_i \in \mathbb{F}$, $i = 1, \ldots, N$, where the notation * denotes the conjugate element, is called *Positive Definite*. In matrix analysis literature, this is the definition of a positive semidefinite matrix. However, since this is a rather cumbersome term and the distinction between positive definite and positive semidefinite matrices is not important in this paper, we employ the term positive definite in the way presented here. Furthermore, the term positive definite was introduced for the first time by Mercer in the kernel context (see [26]). 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*. In the following, we will frequently refer to a positive definite kernel simply as *kernel*.

Next, consider a linear class $\mathcal{H}$ of complex valued functions f defined on a set X. Suppose further, that in $\mathcal{H}$ we can define an inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ with corresponding norm $\| \cdot \|_{\mathcal{H}}$ and that $\mathcal{H}$ is complete with respect to that norm, i.e., $\mathcal{H}$ is a Hilbert space. We call $\mathcal{H}$ a *Reproducing Kernel Hilbert Space (RKHS)*, if for all $y \in X$ the evaluation functional $T_y : \mathcal{H} \to \mathbb{F} : T_y(f) = f(y)$ is a linear continuous (or, equivalently, bounded) operator. If this is true, then by the Riesz's representation theorem, for all $y \in X$ there is a function $g_y \in \mathcal{H}$ such that $T_y(f) = f(y) = \langle f, g_y \rangle_{\mathcal{H}}$. The function $\kappa : X \times X \to \mathbb{F} : \kappa(x, y) = g_y(x)$ is called a *reproducing kernel* of $\mathcal{H}$. It can be easily proved that the function κ is a positive definite kernel.

Alternatively, we can define a RKHS as a Hilbert space $\mathcal{H}$ for which there exists a 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 $\mathcal{H}$.
2) κ has the so called *reproducing property*, i.e.,

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

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

It has been shown (see [27]) that to every positive definite kernel κ there corresponds one and only one class of functions $\mathcal{H}$ with a uniquely determined inner product in it, forming a Hilbert space and admitting κ as a reproducing kernel. In fact, the kernel κ produces the entire space $\mathcal{H}$, i.e., $\mathcal{H} = \overline{\text{span}\{\kappa(x, \cdot) | x \in X\}}^2$. The map $\Phi : X \to \mathcal{H} : \Phi(x) = \kappa(\cdot, x)$ is called the *feature map* of $\mathcal{H}$. 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:

$$\langle af + bg, h \rangle_{\mathcal{H}} = a \langle f, h \rangle_{\mathcal{H}} + b \langle g, h \rangle_{\mathcal{H}},$$
$$\langle f, ag + bh \rangle_{\mathcal{H}} = a^* \langle f, g \rangle_{\mathcal{H}} + b^* \langle f, h \rangle_{\mathcal{H}},$$
$$\langle f, g \rangle_{\mathcal{H}}^* = \langle g, f \rangle_{\mathcal{H}},$$

[2]The overbar denotes the closure of the set.

for all $f, g, h \in \mathcal{H}$, and $a, b \in \mathbb{C}$. In the real case, the condition $\kappa(x, y) = \langle \kappa(\cdot, y), \kappa(\cdot, x) \rangle_{\mathcal{H}}$ may be replaced by $\kappa(x, y) = \langle \kappa(\cdot, x), \kappa(\cdot, y) \rangle_{\mathcal{H}}$. However, since in the complex case the inner product is Hermitian, the aforementioned condition is equivalent to $\kappa(x, y) = (\langle \kappa(\cdot, x), \kappa(\cdot, y) \rangle_{\mathcal{H}})^*$. One of the most important properties of RKHSs is that norm convergence implies pointwise convergence. More precisely, let $\{f_n\}_{n \in \mathbb{N}} \subset \mathcal{H}$ be a sequence such that $\lim_n \|f_n - f\| = 0$, for some $f \in \mathcal{H}$. Then, the continuity of T_x gives $\lim_n f_n(x) = \lim_n T_x(f_n) = T_x(f) = f(x)$, for all $x \in X$.

Although, the underlying theory has been developed by the mathematicians for general complex reproducing kernels and their associated RKHSs, only the real kernels have been considered by the machine learning community. One of the most widely used kernel is the *Gaussian RBF*, i.e.,

$$\kappa_{\sigma, \mathbb{R}^d}(\boldsymbol{x}, \boldsymbol{y}) := \exp\left(-\frac{\sum_{i=1}^{d} (x_i - y_i)^2}{\sigma^2} \right), \qquad (2)$$

defined for $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$, where σ is a free positive parameter. Another popular kernel is the *polynomial kernel*: $\kappa_d(\boldsymbol{x}, \boldsymbol{y}) := (1 + \boldsymbol{x}^T \boldsymbol{y})^d$, for $d \in \mathbb{N}$. Many more can be found in the related literature [7]–[9].

Complex reproducing kernels, that have been extensively studied by the mathematicians, are, among others, the *Szego kernels*, i.e. $\kappa(z, w) = \frac{1}{1 - w^* z}$, for Hardy spaces on the unit disk, and the Bergman kernels, i.e., $\kappa(z, w) = \frac{1}{(1 - w^* z)^2}$, for Bergman spaces on the unit disk, where $|z|, |w| < 1$ [25]. In the following, we discuss another complex kernel that has remained relatively unknown in the Machine Learning and Signal Processing societies.

B. The Complex Gaussian Kernel

Consider the complex valued function

$$\kappa_{\sigma, \mathbb{C}^d}(\boldsymbol{z}, \boldsymbol{w}) := \exp\left(-\frac{\sum_{i=1}^{d} (z_i - w_i^*)^2}{\sigma^2} \right), \qquad (3)$$

defined on $\mathbb{C}^d \times \mathbb{C}^d$, where $\boldsymbol{z}, \boldsymbol{w} \in \mathbb{C}^d$, z_i denotes the i-th component of the complex vector $\boldsymbol{z} \in \mathbb{C}^d$ and $\exp$ is the extended exponential function in the complex domain. It can be shown that $\kappa_{\sigma, \mathbb{C}^d}$ is a complex valued kernel, which we call the *complex Gaussian kernel* with parameter σ. Its restriction $\kappa_\sigma := (\kappa_{\sigma, \mathbb{C}^d})_{|\mathbb{R}^d \times \mathbb{R}^d}$ is the well known *real Gaussian kernel*. An explicit description of the RKHSs of these kernels, together with some important properties can be found in [28].

III. KERNEL LEAST MEAN SQUARE ALGORITHM

In a typical LMS filter the goal is to learn a linear input-output mapping $f : X \to \mathbb{R} : f(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x}$, $X \subset \mathbb{R}^\nu$, based on a sequence of examples $(\boldsymbol{x}(1), d(1)), (\boldsymbol{x}(2), d(2)), \dots, (\boldsymbol{x}(N), d(N))$, so that to minimize the mean square error, $E[|d(n) - \boldsymbol{w}^T \boldsymbol{x}(n)|^2]$. To this end, the gradient descent rationale is employed and at each time instant, $n = 1, 2, \dots, N$, the gradient of the mean square error, i.e., $-2E[e(n)\boldsymbol{x}(n)]$, is estimated via its current measurement, i.e., $\hat{E}[e(n)\boldsymbol{x}(n)] = e(n)\boldsymbol{x}(n)$, where $e(n) = d(n) - \boldsymbol{w}(n-1)^T \boldsymbol{x}(n)$ is the a-priori error at instance

$n = 2, \dots, N$. It takes a few lines of elementary algebra to deduce that the update of the unknown vector parameter is: $\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \mu e(n)\boldsymbol{x}(n)$, where μ is the parameter controlling the step update. If we take the initial value of $\boldsymbol{w}$ as $\boldsymbol{w}(0) = \boldsymbol{0}$, then the repeated application of the update equation yields:

$$\boldsymbol{w}(n) = \mu \sum_{k=1}^{n} e(k)\boldsymbol{x}(k) \qquad (4)$$

Hence, for the filter output at instance n we have:

$$\hat{d}(n) = \boldsymbol{w}(n-1)^T \boldsymbol{x}(n) = \mu \sum_{k=1}^{n-1} e(k)\boldsymbol{x}(k)^T \boldsymbol{x}(n), \quad (5)$$

for $n = 1, 2, \dots, N$. Equation (5) is expressed in terms of inner products only, hence it allows for the application of the kernel trick. Thus, the filter output of the KLMS at instance n is

$$\hat{d}(n) = \langle \boldsymbol{x}(n), \boldsymbol{w}(n-1) \rangle_{\mathcal{H}} = \mu \sum_{k=1}^{n-1} e(k)\kappa(\boldsymbol{x}(n), \boldsymbol{x}(k)), \tag{6}$$

$$\text{while} \qquad \boldsymbol{w}(n) = \mu \sum_{k=1}^{n} e(k)\kappa(\cdot, \boldsymbol{x}(k)), \qquad (7)$$

for $n = 1, 2, \dots, N$.

Another, more formal, way of developing the KLMS is the following. First, we transform the input space X to a high dimensional feature space $\mathcal{H}$, through the (implicit) mapping $\Phi : X \to \mathcal{H}$, $\Phi(\boldsymbol{x}) = \kappa(\cdot, \boldsymbol{x})$. Thus, the training examples become $(\Phi(\boldsymbol{x}(1)), d(1)), \dots, (\Phi(\boldsymbol{x}(N)), d(N))$. We apply the LMS procedure on the transformed data, with the linear filter output $\hat{d}(n) = \langle \Phi(\boldsymbol{x}(n)), \boldsymbol{w} \rangle_{\mathcal{H}}$. The model $\langle \Phi(\boldsymbol{x}), \boldsymbol{w} \rangle_{\mathcal{H}}$ is more representative than the simple $\boldsymbol{w}^T \boldsymbol{x}$, since it includes the nonlinear modeling through the presence of the kernel. The objective now becomes to minimize the cost function $E[|d(n) - \langle \Phi(\boldsymbol{x}(n)), \boldsymbol{w} \rangle_{\mathcal{H}}|^2]$ (see [29]). Using the notion of the Fréchet derivative [29]–[31], which has to be mobilized, since the dimensionality of the RKHS may be infinite, we are able to derive the gradient of the aforementioned cost function with respect to $\boldsymbol{w}$, if we estimate it by its current measurement $|d(n) - \langle \Phi(\boldsymbol{x}(n)), \boldsymbol{w} \rangle|^2$. Thus the respective gradient is $-2e(n)\Phi(\boldsymbol{x}(n))$. It has to be emphasized, that now $\boldsymbol{w}$ is not a vector, but a function, i.e., a point in the linear Hilbert space. It turns out that the update of the KLMS is given by $\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \mu e(n)\Phi(\boldsymbol{x}(n))$, where $e(n) = d(n) - \hat{d}(n)$. From this update, following the same procedure as in LMS and applying the reproducing property, we obtain equations (6) and (7), which are at the core of the KLMS algorithm. More details and the algorithmic implementation may be found in [14].

Note that in a number of attempts to kernelize known algorithms, that are cast in inner products, the kernel trick is, usually, used in a "black box" rationale, without consideration of the problem in the RKH space, in which the (implicit) processing is carried out. Such an approach, often, does not allow for a deeper understanding of the problem, especially if a further theoretical analysis is required. Moreover, in our

case, such a "blind" application of the kernel trick on a standard complex LMS form, can only lead to spaces defined by complex kernels, as it will become clear soon. Complex RKH spaces, that are built around complexification of real kernels, do not result as a direct application of the standard kernel trick.

IV. COMPLEXIFICATION OF REAL REPRODUCING KERNEL HILBERT SPACES

To generalize the kernel adaptive filtering algorithms on complex domains, we need a universal framework regarding complex RKHSs. A first straightforward approach is to use directly a complex RKHS, using one of the complex kernels given in section II. In this section, we present an alternative simple technique called *complexification* of real RKHSs, which 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, e.t.c.).

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 $\mathcal{H}$ be a real RKHS associated with a real kernel κ defined on $X^2 \times X^2$ and let $\langle \cdot, \cdot \rangle_\mathcal{H}$ be its corresponding inner product. Then, every $f \in \mathcal{H}$ 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 $\mathcal{H}^2 = \mathcal{H} \times \mathcal{H}$. It is easy to verify that $\mathcal{H}^2$ is also a Hilbert Space with inner product

$$\langle \boldsymbol{f}, \boldsymbol{g} \rangle_{\mathcal{H}^2} = \langle f_1, g_1 \rangle_\mathcal{H} + \langle f_2, g_2 \rangle_\mathcal{H}, \tag{8}$$

for $\boldsymbol{f} = (f_1, f_2)$, $\boldsymbol{g} = (g_1, g_2)$. Our objective is to enrich $\mathcal{H}^2$ with a complex structure. We address this problem using the complexification of the real RKHS $\mathcal{H}$. To this end, we define the space $\mathbb{H} = \{\boldsymbol{f} = f_1 + if_2; f_1, f_2 \in \mathcal{H}\}$ equipped with the complex inner product:

$$\langle \boldsymbol{f}, \boldsymbol{g} \rangle_\mathbb{H} = \langle f_1, g_1 \rangle_\mathcal{H} + \langle f_2, g_2 \rangle_\mathcal{H} + i\left(\langle f_2, g_1 \rangle_\mathcal{H} - \langle f_1, g_2 \rangle_\mathcal{H}\right),$$

for $\boldsymbol{f} = f_1 + if_2$, $\boldsymbol{g} = g_1 + ig_2$. Hence, $\boldsymbol{f}, \boldsymbol{g} : \mathbb{X} \subseteq \mathbb{C}^\nu \to \mathbb{C}$. It is not difficult to verify that $\mathbb{H}$ is a complex RKHS with kernel κ [25]. We call $\mathbb{H}$ the complexification of $\mathcal{H}$. 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 framework for working on complex RKHSs using this rationale, we need a technique to implicitly map the samples data from the complex input space to the complexified RKHS $\mathbb{H}$. This can be done using the simple rule:

$$\boldsymbol{\Phi}(z) = \boldsymbol{\Phi}(x + iy) = \boldsymbol{\Phi}(x, y) = \Phi(x, y) + i\Phi(x, y), \tag{9}$$

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 $\boldsymbol{\Phi}$ is not the feature map associated with the complex RKHS $\mathbb{H}$. Furthermore, the employed kernel is a real one. Therefore, the algorithms derived using this approach cannot be reproduced, if one blindly applies the kernel trick using any complex kernel. However, observe that:

$$\langle \boldsymbol{\Phi}(z), \boldsymbol{\Phi}(z') \rangle_\mathbb{H} = 2\langle \Phi(x, y), \Phi(x', y') \rangle_\mathcal{H}$$
$$= 2\kappa((x, y), (x', y')).$$

This relation implies that the complexification procedure is equivalent with the following *complexified real kernel trick*: Given an algorithm, which is formulated in terms of complex dot products (i.e, $\boldsymbol{w}^H \boldsymbol{z}$, where $\boldsymbol{z} = \boldsymbol{x} + i\boldsymbol{y}$, $\boldsymbol{w} = \boldsymbol{w}_1 + i\boldsymbol{w}_2$), one can construct an alternative algorithm by replacing each one of the complex dot products with a positive definite *real kernel* κ, with arguments the extended real vectors of $\boldsymbol{z}$ and $\boldsymbol{w}$ (i.e., $\kappa((\boldsymbol{x}, \boldsymbol{y}), (\boldsymbol{w}_2, \boldsymbol{w}_2))$).

V. WIRTINGER'S CALCULUS ON $\mathbb{C}$

Wirtinger's calculus [15] is enjoying increasing popularity in the signal processing community mainly in the context of complex adaptive filtering [16]–[23], as a means to compute, in an elegant way, gradients of real valued cost functions that are defined on complex domains ($\mathbb{C}^\nu$). The Cauchy-Riemann conditions dictate that such functions are not holomorphic (except from the case where the function is a constant) and therefore the complex derivative cannot be used. Instead, if we consider that the cost function is defined on a Euclidean domain with a double dimensionality ($\mathbb{R}^{2\nu}$), then the real derivatives may be employed. The price of this approach is that the computations may become cumbersome and tedious. Wirtinger's calculus provides an alternative equivalent formulation, that is based on simple rules and principles and which bears a great resemblance to the rules of the standard complex derivative. In this section, we present the main notions of Wirtinger's calculus for functions defined on complex domains. These ideas are, subsequently, extended in section VI to include the case of general complex Hilbert spaces.

Let $f : \mathbb{C} \to \mathbb{C}$ be a complex function defined on $\mathbb{C}$. Obviously, such a function may be regarded as either defined on $\mathbb{R}^2$ or $\mathbb{C}$ (i.e., $f(z) = f(x + iy) = f(x, y)$). Furthermore, it may be regarded as either a complex valued function, $f(x, y) = u(x, y) + iv(x, y)$ or as a vector valued function $f(x, y) = (u(x, y), v(x, y))$. We will say that f is *differentiable in the real sense* if u and v are differentiable. It turns out that, when the complex structure is considered, the real derivatives may be described using an equivalent and more elegant formulation, which bears a surprising resemblance with the complex derivative. In fact, if the function f is *differentiable in the complex sense* (i.e. the complex derivative exists), the developed derivatives coincide with the complex ones. Although this methodology is known for some time in the German speaking countries and it has been applied to practical applications [32], [33], only recently has attracted the attention of the signal processing community, mostly in the context of works that followed Picinbono's paper on widely linear estimation filters [16].

The *Wirtinger's derivative* (or *W-derivative* for short) of f at a point c is defined as follows

$$\frac{\partial f}{\partial z}(c) = \frac{1}{2}\left(\frac{\partial f}{\partial x}(c) - i\frac{\partial f}{\partial y}(c)\right)$$
$$= \frac{1}{2}\left(\frac{\partial u}{\partial x}(c) + \frac{\partial v}{\partial y}(c)\right) + \frac{i}{2}\left(\frac{\partial v}{\partial x}(c) - \frac{\partial u}{\partial y}(c)\right). \tag{10}$$

The *conjugate Wirtinger's derivative* (or *CW-derivative* for short) of f at c is defined by:

$$\frac{\partial f}{\partial z^*}(c) = \frac{1}{2}\left(\frac{\partial f}{\partial x}(c) + i\frac{\partial f}{\partial y}(c)\right)$$
$$= \frac{1}{2}\left(\frac{\partial u}{\partial x}(c) - \frac{\partial v}{\partial y}(c)\right) + \frac{i}{2}\left(\frac{\partial v}{\partial x}(c) + \frac{\partial u}{\partial y}(c)\right). \quad (11)$$

The following properties can be proved [21], [34], [35]:

1) If f has a Taylor series expansion with respect to z (i.e., it is holomorphic) around c, then $\frac{\partial f}{\partial z^*}(c) = 0$.
2) If f has a Taylor series expansion with respect to z^* around c, then $\frac{\partial f}{\partial z}(c) = 0$.
3) $\left(\frac{\partial f}{\partial z}(c)\right)^* = \frac{\partial f^*}{\partial z^*}(c)$.
4) $\left(\frac{\partial f}{\partial z^*}(c)\right)^* = \frac{\partial f}{\partial z^*}(c)$.
5) Linearity: If f, g are differentiable in the real sense at c and $\alpha, \beta \in \mathbb{C}$, then

$$\frac{\partial(\alpha f + \beta g)}{\partial z}(c) = \alpha\frac{\partial f}{\partial z}(c) + \beta\frac{\partial g}{\partial z}(c),$$
$$\frac{\partial(\alpha f + \beta g)}{\partial z^*}(c) = \alpha\frac{\partial f}{\partial z^*}(c) + \beta\frac{\partial g}{\partial z^*}(c)$$

6) Product Rule: If f, g are differentiable in the real sense at c, then

$$\frac{\partial(f \cdot g)}{\partial z}(c) = \frac{\partial f}{\partial z}(c)g(c) + f(c)\frac{\partial g}{\partial z}(c),$$
$$\frac{\partial(f \cdot g)}{\partial z^*}(c) = \frac{\partial f}{\partial z^*}(c)g(c) + f(c)\frac{\partial g}{\partial z^*}(c).$$

7) Division Rule: If f, g are differentiable in the real sense at c and $g(c) \neq 0$, then

$$\frac{\partial(\frac{f}{g})}{\partial z}(c) = \frac{\frac{\partial f}{\partial z}(c)g(c) - f(c)\frac{\partial g}{\partial z}(c)}{g^2(c)},$$
$$\frac{\partial(\frac{f}{g})}{\partial z^*}(c) = \frac{\frac{\partial f}{\partial z^*}(c)g(c) - f(c)\frac{\partial g}{\partial z^*}(c)}{g^2(c)}.$$

8) Chain Rule: If f is differentiable in the real sense at c and g is differentiable in the real sense at $f(c)$, then

$$\frac{\partial g \circ f}{\partial z}(c) = \frac{\partial g}{\partial z}(f(c))\frac{\partial f}{\partial z}(c) + \frac{\partial g}{\partial z^*}(f(c))\frac{\partial f^*}{\partial z}(c),$$
$$\frac{\partial g \circ f}{\partial z^*}(c) = \frac{\partial g}{\partial z}(f(c))\frac{\partial f}{\partial z^*}(c) + \frac{\partial g}{\partial z^*}(f(c))\frac{\partial f^*}{\partial z^*}(c).$$

In view of the aforementioned properties, one might easily compute the W and CW derivatives of any complex function f, which is written in terms of z and z^*, following the following simple tricks:

- *To compute the W-derivative of a function f, which is expressed in terms of z and z^*, apply the usual differentiation rules considering z^* as a constant.*
- *To compute the CW-derivative of a function f, which is expressed in terms of z and z^*, apply the usual differentiation rules considering z as a constant.*

Note that any complex function $f(z)$, which is differentiable in the real sense, can be cast in terms of z and z^*. For example,

if the function $f(z) = f(x+iy) = f(x, y)$ is given in terms of x and y, replacing x by $(z + z^*)/2$ and y by $(z - z^*)/2$ gives the result. It should be emphasized, that these statements must be regarded as a simple computational trick rather than as a rigorous mathematical rule. This trick works well due to the aforementioned properties. Nonetheless, special care should be considered whenever these tricks are applied. For example, given the function $f(z) = |z|^2$, we might conclude that $\frac{\partial f}{\partial z^*} = 0$, since if we consider z as a constant, then $f(z)$ is also a constant. However, one might argue that since there isn't any rule regarding the complex norm, this rationale leads to an error. Undeniably, if one recasts f as $f(z) = zz^*$, then one concludes that $\frac{\partial f}{\partial z^*} = z$ and $\frac{\partial f}{\partial z} = z^*$. Similar rules and principles hold for functions defined on $\mathbb{C}^\nu$ [34].

VI. EXTENSION OF WIRTINGER'S CALCULUS TO GENERAL HILBERT SPACES

To apply minimization algorithms on real valued operators defined on complex RKHSs, we need to compute the associated gradients. To this end, in this section, we generalize the main ideas and results of Wirtinger's calculus on general Hilbert spaces. We begin with a brief review of the *Fréchet derivative*, which generalizes differentiability to Hilbert spaces and which will be the basis for our discussion.

A. Fréchet Derivatives

Since Fréchet differentiability is not the mainstream of mathematical tools used in the Signal Processing and Machine Learning communities, we give here some basic definitions for the sake of clarity. Consider a Hilbert space H over the field F (typically $\mathbb{R}$ or $\mathbb{C}$). The operator $\boldsymbol{T} : H \to F^\nu$ is said to be *Fréchet differentiable* at f_0, if there exists a linear continuous operator $\boldsymbol{W} = (W_1, W_2, \ldots, W_\nu) : H \to \mathbb{F}^\nu$ such that

$$\lim_{\|h\|_H \to 0} \frac{\|\boldsymbol{T}(f_0 + h) - \boldsymbol{T}(f_0) - \boldsymbol{W}(h)\|_{F^\nu}}{\|h\|_H} = 0, \quad (12)$$

where $\|\cdot\|_H = \sqrt{\langle\cdot,\cdot\rangle_H}$ is the induced norm of the corresponding Hilbert Space. Note that F^ν is considered as a Banach space under the Euclidean norm. The linear operator $\boldsymbol{W}$ is called the *Fréchet derivative* and is usually denoted by $d\boldsymbol{T}(f_0) : H \to F^\nu$. Observe that this definition is valid not only for Hilbert spaces, but for general Banach spaces too. However, since we are mainly interested at Hilbert spaces, we present the main ideas in this context. It can be proved that if such a linear continuous operator $\boldsymbol{W}$ can be found, then it is unique (i.e., the derivative is unique) [30]. In the special case where $\nu = 1$ (i.e., the operator $\boldsymbol{T}$ takes values on F) using the *Riesz's representation* theorem, we may replace the linear continuous operator $\boldsymbol{W}$ with an inner product. Therefore, the operator $T : H \to F$ is said to be *Fréchet differentiable* at f_0, iff there exists a $w \in H$, such that

$$\lim_{\|h\|_H \to 0} \frac{T(f_0 + h) - T(f_0) - \langle h, w\rangle_H}{\|h\|_H} = 0, \quad (13)$$

where $\langle\cdot,\cdot\rangle_H$ is the dot product of the Hilbert space H and $\|\cdot\|_H$ is the induced norm. The element w^* is usually called the gradient of T at f_0 and it is denoted by $w^* = \nabla T(f_0)$.

For a general vector valued operator $\boldsymbol{T} = (T_1, \ldots, T_\nu) : H \to F^\nu$, we may easily derive that if $\boldsymbol{T}$ is differentiable at f_0, then T_ι is differentiable at f_0, for all $\iota = 1, 2, \ldots, \nu$, and that

$$d\boldsymbol{T}(f_0)(h) = \begin{pmatrix} \langle h, \nabla T_1(f_0)^* \rangle_H \\ \vdots \\ \langle h, \nabla T_\nu(f_0)^* \rangle_H \end{pmatrix}. \qquad (14)$$

To prove this claim, consider that since $\boldsymbol{T}$ is differentiable, there exists a continuous linear operator $\boldsymbol{W}$ such that

$$\lim_{\|h\|_H \to 0} \frac{\|\boldsymbol{T}(f_0 + h) - \boldsymbol{T}(f_0) - \boldsymbol{W}(h)\|_{F^\nu}}{\|h\|_H} = 0 \Leftrightarrow$$
$$\lim_{\|h\|_H \to 0} \left(\sum_{\iota=1}^\nu \frac{|T_\iota(f_0 + h) - T_\iota(f_0) - W_\iota(h)|_F^2}{\|h\|_H^2} \right) = 0,$$

for all $\iota = 1, \ldots, \nu$. Thus,

$$\lim_{\|h\|_H \to 0} \left(\frac{T_\iota(f_0 + h) - T_\iota(f_0) - W_\iota(h)}{\|h\|_H} \right) = 0,$$

for all $\iota = 1, 2, \nu$. The Riesz's representation theorem dictates that since W_ι is a continuous linear operator, there exists $w_\iota \in H$, such that $W_\iota(h) = \langle h, w_\iota \rangle_H$, for all $\iota = 1, \ldots, \nu$. This proves that T_ι is differentiable at f_0 and that $w_\iota^* = \nabla T_\iota(f_0)$, thus equation (14) holds. The converse is proved similarly.

The notion of *Fréchet differentiability* may be extended to include also partial derivatives. Consider the operator $T : H^\mu \to F$ defined on the Hilbert space H^μ with corresponding inner product:

$$\langle \boldsymbol{f}, \boldsymbol{g} \rangle_{H^\mu} = \sum_{\iota=1}^\mu \langle f_\iota, g_\iota \rangle_H,$$

where $\boldsymbol{f} = (f_1, f_2, \ldots f_\mu)$, $\boldsymbol{g} = (g_1, g_2, \ldots g_\mu)$. $T(\boldsymbol{f})$ is said to be *Fréchet differentiable* at $\boldsymbol{f}_0$ with respect to f_ι, iff there exists a $w \in H$, such that

$$\lim_{\|h\|_H \to 0} \frac{T(\boldsymbol{f}_0 + [h]_\iota) - T(\boldsymbol{f}_0) - \langle [h]_\iota, w \rangle_H}{\|h\|_H} = 0, \quad (15)$$

where $[h]_\iota = (0, 0, \ldots, 0, h, 0, \ldots, 0)$, is the element of H^μ with zero entries everywhere, except at place ι. The element w^* is called the gradient of T at $\boldsymbol{f}_0$ with respect to f_ι and it is denoted by $w^* = \nabla_\iota T(\boldsymbol{f}_0)$. The Fréchet partial derivative at $\boldsymbol{f}_0$ with respect to f_ι is denoted by $\frac{\partial T}{\partial f_\iota}(\boldsymbol{f}_0)$, $\frac{\partial T}{\partial f_\iota}(\boldsymbol{f}_0)(h) = \langle [h]_\iota, w \rangle_{\mathbb{H}}$.

Although it will not be used here, it is interesting to note, that it is also possible to define Fréchet derivatives of higher order and a corresponding Taylor's series expansion. In this context, the n-th Fréchet derivative of $\boldsymbol{T}$ at $\boldsymbol{f}_0$, denoted as $d^n \boldsymbol{T}(\boldsymbol{f}_0)$, is a multilinear[3] map. If $\boldsymbol{T}$ has Fréchet derivatives of any order, it can be expanded as a Taylor series [36], i.e.,

$$\boldsymbol{T}(\boldsymbol{f}_0 + \boldsymbol{h}) = \sum_{n=0}^\infty \frac{1}{n!} d^n \boldsymbol{T}(\boldsymbol{f}_0)(\boldsymbol{h}, \boldsymbol{h}, \ldots, \boldsymbol{h}). \qquad (16)$$

In relative literature the term $d^n \boldsymbol{T}(\boldsymbol{c})(\boldsymbol{h}, \boldsymbol{h}, \ldots, \boldsymbol{h})$ is often replaced by $d^n \boldsymbol{T}(\boldsymbol{c}) \cdot \boldsymbol{h}^n$, which it denotes that the multilinear map $d^n \boldsymbol{T}(\boldsymbol{c})$ is applied to $(\boldsymbol{h}, \boldsymbol{h}, \ldots, \boldsymbol{h})$.

[3] A function is called multilinear, if it is linear in each variable.

B. Complex Hilbert spaces

Let $\mathcal{H}$ be a real Hilbert space with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ and $\mathcal{H}^2$, $\mathbb{H}$ the Hilbert spaces defined as shown in section IV. In the following, the complex structure of $\mathbb{H}$ will be used to derive derivatives similar to the ones obtained from Wirtinger's calculus on $\mathbb{C}$.

Consider the function $\boldsymbol{T} : A \subseteq \mathbb{H} \to \mathbb{C}$, $\boldsymbol{T}(\boldsymbol{f}) = \boldsymbol{T}(u_{\boldsymbol{f}} + iv_{\boldsymbol{f}}) = T_r(u_{\boldsymbol{f}}, v_{\boldsymbol{f}}) + iT_i(u_{\boldsymbol{f}}, v_{\boldsymbol{f}})$, defined on an open subset A of $\mathbb{H}$, where $u_{\boldsymbol{f}}, v_{\boldsymbol{f}} \in \mathcal{H}$ and T_r, T_i are real valued functions defined on $\mathcal{H}^2$. Any such function, $\boldsymbol{T}$, may be regarded as defined either on a subset of $\mathbb{H}$, or on a subset of $\mathcal{H}^2$. Moreover, $\boldsymbol{T}$ 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:

$$\boldsymbol{T}(\boldsymbol{f}) = \boldsymbol{T}(u_{\boldsymbol{f}} + iv_{\boldsymbol{f}}) = T_r(u_{\boldsymbol{f}}, v_{\boldsymbol{f}}) + iT_i(u_{\boldsymbol{f}}, v_{\boldsymbol{f}}),$$
$$\boldsymbol{T}(\boldsymbol{f}) = (T_r(u_{\boldsymbol{f}}, v_{\boldsymbol{f}}), T_i(u_{\boldsymbol{f}}, v_{\boldsymbol{f}})).$$

In the following, we will often change the notation according to the specific problem and consider any element of $\boldsymbol{f} \in \mathbb{H}$ defined either as $\boldsymbol{f} = u_{\boldsymbol{f}} + iv_{\boldsymbol{f}} \in \mathbb{H}$, or as $\boldsymbol{f} = (u_{\boldsymbol{f}}, v_{\boldsymbol{f}}) \in \mathcal{H}^2$. In a similar manner, any complex number may be regarded as either an element of $\mathbb{C}$, or as an element of $\mathbb{R}^2$. We say that $\boldsymbol{T}$ is *Fréchet complex differentiable* at $\boldsymbol{c} \in \mathbb{H}$, if there exists $\boldsymbol{w} \in \mathbb{H}$ such that:

$$\lim_{\|\boldsymbol{h}\|_{\mathbb{H}} \to 0} \frac{\boldsymbol{T}(\boldsymbol{c} + \boldsymbol{h}) - \boldsymbol{T}(\boldsymbol{c}) - \langle \boldsymbol{h}, \boldsymbol{w} \rangle_{\mathbb{H}}}{\|\boldsymbol{h}\|_{\mathbb{H}}} = 0. \qquad (17)$$

Then $\boldsymbol{w}^*$ is called the *complex gradient* of $\boldsymbol{T}$ at $\boldsymbol{c}$ and it is denoted as $\boldsymbol{w}^* \equiv \nabla \boldsymbol{T}(\boldsymbol{c})$. The Fréchet complex derivative of $\boldsymbol{T}$ at $\boldsymbol{c}$ is denoted as $d\boldsymbol{T}(\boldsymbol{c})(\boldsymbol{h}) = \langle \boldsymbol{h}, \boldsymbol{w} \rangle_{\mathbb{H}}$. This definition, although similar with the typical Fréchet derivative, exploits the complex structure of $\mathbb{H}$. More specifically, the complex inner product, that appears in the definition, forces a great deal of structure on $\boldsymbol{T}$. Similarly to the case of ordinary complex functions, it is this simple fact that gives birth to all the important strong properties of the complex derivative. For example, it can be proved, that if $d\boldsymbol{T}(\boldsymbol{c})$ exists, then so does $d^n \boldsymbol{T}(\boldsymbol{c})$, for $n \in \mathbb{N}$. If $\boldsymbol{T}$ is differentiable at any $\boldsymbol{c} \in \mathbb{A}$, $\boldsymbol{T}$ is called *Fréchet holomorphic* in A, or *Fréchet complex analytic* in A, in the sense that it can be expanded as a Taylor series, i.e.,

$$\boldsymbol{T}(\boldsymbol{c} + \boldsymbol{h}) = \sum_{n=0}^\infty \frac{1}{n!} d^n \boldsymbol{T}(\boldsymbol{c})(\boldsymbol{h}, \boldsymbol{h}, \ldots, \boldsymbol{h}). \qquad (18)$$

The proof of this statement is out of the scope of this paper. The interested reader may dig deeper on this subject by referring to [36]. We begin our study by exploring the relations between the complex Fréchet derivative and the real Fréchet derivatives. In the following, we will say that $\boldsymbol{T}$ is *Fréchet differentiable in the complex sense*, if the complex derivative exists, and that $\boldsymbol{T}$ is *Fréchet differentiable in the real sense*, if its real Fréchet derivative exists (i.e., $\boldsymbol{T}$ is regarded as a vector valued operator $\boldsymbol{T} : \mathcal{H}^2 \to \mathcal{H}^2$). Similarly, the expression "$\boldsymbol{T}$ is *Fréchet complex analytic at $\boldsymbol{c}$*" means that $\boldsymbol{T}$ is Fréchet complex analytic at a neighborhood around $\boldsymbol{c}$. We will say that $\boldsymbol{T}$ is *Fréchet real analytic*, when both T_r and T_i have a Taylor's series expansion in the real sense.

Proposition VI.1. *Let $T : A \subset \mathbb{H} \to \mathbb{C}$ be an operator such that $T(f) = T(u_f + iv_f) = T(u_f, v_f) = T_r(u_f, v_f) + iT_i(u_f, v_f)$. If the Fréchet complex derivative of T at a point $c \in A$ (i.e., $dT(c) : \mathbb{H} \to \mathbb{C}$) exists, then T_r and T_i are differentiable at the point $c = (c_1, c_1) = c_1 + ic_2$, where $c_1, c_2 \in \mathcal{H}$. Furthermore,*

$$\nabla_u T_r(c_1, c_2) = \nabla_v T_i(c_1, c_2), \nabla_v T_r(c_1, c_2) = -\nabla_u T_i(c_1, c_2). \tag{19}$$

Equations (19) are the *Cauchy Riemann conditions* with respect to the Fréchet notion of differentiability. Similar to the simple case of complex valued functions, they provide a necessary and sufficient condition, for a complex operator T, that is defined on $\mathbb{H}$, to be differentiable in the complex sense, provided that T is differentiable in the real sense. This is explored in the following proposition.

Proposition VI.2. *If the operator $T : A \subseteq \mathbb{H} \to \mathbb{C}$, $T(f) = T_r(f) + iT_i(f)$, where $f = u_f + iv_f$, is Fréchet differentiable in the real sense at a point $(c_1, c_2) \in \mathcal{H}^2$ and the Fréchet Cauchy-Riemann conditions hold:*

$$\nabla_u T_r(c_1, c_2) = \nabla_v T_i(c_1, c_2), \nabla_v T_r(c_1, c_2) = -\nabla_u T_i(c_1, c_2), \tag{20}$$

then T is differentiable in the complex sense at the point $c = (c_1, c_2) = c_1 + c_2 i \in \mathbb{H}$.

Proof: see Appendix A. ∎

If the Fréchet Cauchy Riemann conditions are not satisfied for an operator T, then the Fréchet complex derivative does not exist and the function cannot be expressed in terms of h, as in the case of Fréchet complex differentiable functions (see equation 18). Nevertheless, if T is Fréchet differentiable in the real sense (i.e., T_r and T_i are Fréchet differentiable), we may still find a form of Taylor's series expansion by utilizing the extension of Wirtinger's calculus. It can be shown (see the proof of proposition VI.2 in Appendix A), that:

$$T(c + h) = T(c) + \frac{1}{2} \left\langle h, (\nabla_u T(c) - i\nabla_v T(c))^* \right\rangle_{\mathbb{H}} \tag{21}$$
$$+ \frac{1}{2} \left\langle h^*, (\nabla_u T(c) + i\nabla_v T(c))^* \right\rangle_{\mathbb{H}} + o(\|h\|_{\mathbb{H}}).$$

One may notice that in this case the associated Taylor's expansion is casted in terms of both h and h^*. This can be generalized for higher order Taylor's expansion formulas by following the same rationale. Observe also that, if T is Fréchet complex differentiable, this relation degenerates (due to the Cauchy Riemann conditions) to the respective Taylor's expansion formula (i.e., (18)). In this context, the following definitions come naturally.

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

$$\nabla_f T(c) = \frac{1}{2} \left(\nabla_1 T(c) - i\nabla_2 T(c) \right)$$
$$= \frac{1}{2} \left(\nabla_u T_r(c) + \nabla_v T_i(c) \right) \tag{22}$$
$$+ \frac{i}{2} \left(\nabla_u T_i(c) - \nabla_v T_r(c) \right),$$

and the *Fréchet Wirtinger's derivative* (or *W-derivative*) as $\frac{\partial T}{\partial f}(c) : \mathbb{H} \to \mathbb{C}$, such that $\frac{\partial T}{\partial f}(c)(h) = \langle h, \nabla_f T(c)^* \rangle_{\mathbb{H}}$. Consequently, the *Fréchet conjugate Wirtinger's gradient* (or *CW-gradient* for short) and the *Fréchet conjugate Wirtinger's derivative* (or *CW-derivative*) of T at c are defined by:

$$\nabla_{f^*} T(c) = \frac{1}{2} \left(\nabla_1 T(c) + i\nabla_2 T(c) \right) \tag{23}$$
$$= \frac{1}{2} \left(\nabla_u T_r(c) - \nabla_v T_i(c) \right)$$
$$+ \frac{i}{2} \left(\nabla_u T_i(c) + \nabla_v T_r(c) \right),$$

and $\frac{\partial T}{\partial f^*}(c) : \mathbb{H} \to \mathbb{C}$, such that $\frac{\partial T}{\partial f^*}(c)(h) = \langle h, (\nabla_{f^*} T(c))^* \rangle_{\mathbb{H}}$. Note, that both the W-derivative and the CW-derivative exist, if T is Fréchet differentiable in the real sense. In view of these new definitions, equation (21) may now be recast as follows

$$T(c + h) = T(c) + \left\langle h, (\nabla_f T(c))^* \right\rangle_{\mathbb{H}} \tag{24}$$
$$+ \left\langle h^*, (\nabla_{f^*} T(c))^* \right\rangle_{\mathbb{H}} + o(\|h\|_{\mathbb{H}}). \tag{25}$$

From these definitions, several properties can be derived:

1) If $T(f)$ is f-holomorphic at c (i.e., it has a Taylor series expansion with respect to f at c), then its Fréchet W-derivative at c degenerates to the standard Fréchet complex derivative and its Fréchet CW-derivative vanishes, i.e., $\nabla_{f^*} T(c) = 0$.

2) If $T(f)$ is f^*-holomorphic at c (i.e., it has a Taylor series expansion with respect to f^* at c), then $\nabla_f T(c) = 0$.

3) The first order Taylor expansion around $f \in \mathbb{H}$ is given by

$$T(f + h) = T(f) + \left\langle h, (\nabla_f T(f))^* \right\rangle_{\mathbb{H}}$$
$$+ \left\langle h^*, (\nabla_{f^*} T(f))^* \right\rangle_{\mathbb{H}}.$$

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

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

6) Linearity: If $T, S : \mathbb{H} \to \mathbb{C}$ are Fréchet differentiable in the real sense at $c \in \mathbb{H}$ and $\alpha, \beta \in \mathbb{C}$, then

$$\nabla_f (\alpha T + \beta S)(c) = \alpha \nabla_f T(c) + \beta \nabla_f S(c)$$
$$\nabla_{f^*} (\alpha T + \beta S)(c) = \alpha \nabla_{f^*} T(c) + \beta \nabla_{f^*} S(c).$$

A complete list of the derived properties, together with the proofs of the most important ones, are given in Appendix B.

An important consequence of the previous properties is that if T is a real valued operator defined on $\mathbb{H}$, then $(\nabla_f T(c))^* = \nabla_{f^*} T(c)$, and its first order Taylor's expansion is given by:

$$T(f + h) = T(f) + \langle h, (\nabla_f T(f))^* \rangle_{\mathbb{H}} + \langle h^*, (\nabla_{f^*} T(f))^* \rangle_{\mathbb{H}}$$
$$= T(f) + \langle h, \nabla_{f^*} T(f) \rangle_{\mathbb{H}} + (\langle h, \nabla_{f^*} T(f) \rangle_{\mathbb{H}})^*$$
$$= T(f) + 2 \cdot \Re \left[\langle h, \nabla_{f^*} T(f) \rangle_{\mathbb{H}} \right].$$

However, in view of the Cauchy Riemann inequality we have:

$$\Re \left[\langle h, \nabla_{f^*} T(f) \rangle_{\mathbb{H}} \right] \le |\langle h, \nabla_{f^*} T(f) \rangle_{\mathbb{H}}|$$
$$\le \|h\|_{\mathbb{H}} \cdot \|\nabla_{f^*} T(f)\|_{\mathbb{H}}.$$

The equality in the above relationship holds if $h \upuparrows \nabla_{f^*} T$ (where the notation $\upuparrows$ denotes that h and $\nabla_{f^*} T$ have the same direction, i.e., there is a $\lambda > 0$, such that $h = \lambda \nabla_{f^*} T$). Hence, the direction of increase of T is $\nabla_{f^*} T(f)$. Therefore, any gradient descent based algorithm minimizing $T(f)$ is based on the update scheme:

$$f_n = f_{n-1} - \mu \cdot \nabla_{f^*} T(f_{n-1}). \qquad (26)$$

Assuming differentiability of T, a standard result from Fréchet real calculus states that a necessary condition for a point c to be an optimum (in the sense that $T(f)$ is minimized or maximized) is that this point is a stationary point of T, i.e., the Fréchet partial derivatives of T at c vanish. In the context of Wirtinger's calculus, we have the following obvious corresponding result.

Proposition VI.3. *If the function $T : A \subseteq \mathbb{H} \to \mathbb{R}$ is Fréchet differentiable at c in the real sense, then a necessary condition for a point c to be a local optimum (in the sense that $T(c)$ is minimized or maximized) is that either the Fréchet W, or the Fréchet CW derivative vanishes.*

Proof: Observe that if T is real valued, the Wirtinger derivatives take the form $\nabla_f T(c) = \frac{1}{2}(\nabla_u T(c) - i\nabla_v T(c))$ and $\nabla_{f^*} T(c) = \frac{1}{2}(\nabla_u T(c) + i\nabla_v T(c))$. If c is a local optimum of T then $\nabla_u T(c) = \nabla_v T(c) = 0$ and thus $\nabla_f T(c) = \nabla_{f^*} T(c) = 0$. Note, that for real valued functions the W and the CW derivatives constitute a conjugate pair. Thus, if the W derivative vanishes, then the CW derivative vanishes too. The converse is also true. This completes the proof. ∎

VII. COMPLEX KERNEL LEAST MEAN SQUARES - CKLMS

In order to illustrate how the proposed framework may be applied to problems of complex signal processing, we present two realizations of the Kernel Least Mean Squares (KLMS) algorithm for complex data. The first scheme (CKLMS1) employs the complexification of real reproducing kernels (see section IV), while the second one uses pure complex kernels (CKLMS2). Wirtinger's calculus is exploited in both cases to compute the necessary gradient updates.

A. Complex KLMS via complexification of real kernels - CKLMS1

Consider the sequence of examples $(z(1), d(1))$, $(z(2), d(2))$, ..., $(z(N), d(N))$, where $d(n) \in \mathbb{C}$, $z(n) \in V \subset \mathbb{C}^\nu$, $z(n) = x(n) + iy(n)$, $x(n), y(n) \in \mathbb{R}^\nu$, for $n = 1, \ldots, N$. Consider, also, a real reproducing kernel κ defined on $X \times X$, $X \subseteq \mathbb{R}^{2\nu}$, and let $\mathcal{H}$ be the corresponding RKHS. We map the points $z(n)$ to the RKHS $\mathbb{H}$ ($\mathbb{H}$ is constructed as explained in section IV) using the mapping $\mathbf{\Phi}$:

$$\mathbf{\Phi}(z(n)) = \Phi(z(n)) + i\Phi(z(n))$$
$$= \kappa(\cdot, (x(n), y(n))) + i \cdot \kappa(\cdot, (x(n), y(n))),$$

for $n = 1, \ldots, N$, where Φ is the feature map of $\mathcal{H}$. The objective of the complex Kernel LMS is to design a filter, w, with desired response $\hat{d}(n) = \langle \mathbf{\Phi}(z(n)), w \rangle_{\mathbb{H}}$, so that to minimize $E[\mathcal{L}_n(w)]$, where

$$\mathcal{L}_n(w) = |e(n)|^2 = |d(n) - \langle \mathbf{\Phi}(z(n)), w \rangle_{\mathbb{H}}|^2$$
$$= (d(n) - \langle \mathbf{\Phi}(z(n)), w \rangle_{\mathbb{H}})(d(n) - \langle \mathbf{\Phi}(z(n)), w \rangle_{\mathbb{H}})^*$$
$$= (d(n) - \langle w^*, \mathbf{\Phi}^*(z(n)) \rangle_{\mathbb{H}})(d(n)^* - \langle w, \mathbf{\Phi}(z(n)) \rangle_{\mathbb{H}}),$$

at each instance n. We then apply the complex LMS to the transformed data, estimating the mean square error by its current measurement $\hat{E}[\mathcal{L}_n(w)] = \mathcal{L}_n(w)$, using the rules of Wirtinger's calculus to compute the CW gradient, i.e., $\nabla_{w^*} \mathcal{L}_n(w) = -e(n)^* \cdot \mathbf{\Phi}(z(n))$. Therefore the CKLMS1 update rule becomes:

$$w(n) = w(n-1) + \mu e(n)^* \cdot \mathbf{\Phi}(z(n)), \qquad (27)$$

where $w(n)$ denotes the estimate at iteration n.

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

$$w(n) = w(n-1) + \mu e(n)^* \mathbf{\Phi}(z(n))$$
$$= w(n-2) + \mu e(n-1)^* \mathbf{\Phi}(z(n-1))$$
$$+ \mu e(n)^* \mathbf{\Phi}(z(n))$$
$$= \mu \sum_{k=1}^{n} e(k)^* \mathbf{\Phi}(z(k)). \qquad (28)$$

Thus, the filter output at iteration n becomes:

$$\hat{d}(n) = \langle \mathbf{\Phi}(z(n)), w(n-1) \rangle_{\mathbb{H}}$$
$$= \mu \sum_{k=1}^{n-1} e(k) \langle \mathbf{\Phi}(z(n)), \mathbf{\Phi}(z(k)) \rangle_{\mathbb{H}}$$
$$= 2\mu \sum_{k=1}^{n-1} \Re[e(k)] \kappa(z(n), z(k))$$
$$+ 2\mu \cdot i \sum_{k=1}^{n-1} \Im[e(k)] \kappa(z(n), z(k)), \qquad (29)$$

where the evaluation of the kernel is done by replacing the complex vectors $z(n)$, of $\mathbb{C}^\nu$ with the corresponding real vectors of $\mathbb{R}^{2\nu}$, i.e., $z(n) = (x(n), y(n))$.

It can readily be shown that, since the CKLMS1 is the complex LMS in RKHS, the important properties of the LMS (convergence in the mean, misadjustment, e.t.c.) carry over to CKLMS1. Furthermore, we may also define a normalized version, which we call *Normalized Complex Kernel LMS* (NCKLMS1). The weight-update of the NCKLMS1 is given by:

$$w(n) = w(n-1) + \frac{\mu}{2 \cdot \kappa(z(n), z(n))} e(n)^* \mathbf{\Phi}(z(n))$$

The NCKLMS1 algorithm is summarized in Algorithm 1. We should emphasize that this formulation of the complex KLMS cannot be derived following the usual "black box" rationale of the kernel trick, as it has already been pointed out in section IV. The complexified real kernel trick can be used instead.

One might think, that modeling the desired response as $\hat{d}(n) = \langle w(n-1), \mathbf{\Phi}(z(n)) \rangle_{\mathbb{H}}$, provides an alternative formulation for the CKLMS1 algorithm. In this case, the CW gradient of the instantaneous square error is given by

Algorithm 1 Normalized Complex Kernel LMS with complexification of real kernels (NCKLMS1)

INPUT: $(\boldsymbol{z}(1), d(1)), \ldots, (\boldsymbol{z}(N), d(N))$
OUTPUT: The expansion
$\boldsymbol{w} = \sum_{k=1}^{N} a(k)\kappa(\cdot, \boldsymbol{z}(k)) + i \cdot \sum_{k=1}^{N} b(k)\kappa(\cdot, \boldsymbol{z}(k)).$

Initialization: Set $\boldsymbol{a} = \{\}$, $\boldsymbol{b} = \{\}$, $\boldsymbol{Z} = \{\}$ (i.e., $\boldsymbol{w} = \boldsymbol{0}$). Select the step parameter μ and the kernel κ.

   for $n = 1 : N$ **do**
      Compute the filter output:

$$\hat{d}(n) = \sum_{k=1}^{n-1} (a(k) + b(k)) \cdot \kappa(\boldsymbol{z}(n), \boldsymbol{z}(k))$$
$$+ i \sum_{k=1}^{n-1} (a(k) - b(k)) \cdot \kappa(\boldsymbol{z}(n), \boldsymbol{z}(k)).$$

      Compute the error: $e(n) = d(n) - \hat{d}(n)$.
      $\gamma = 2\kappa(\boldsymbol{z}(n), \boldsymbol{z}(n))$.
      $a(n) = \mu(\Re[e(n)] + \Im[e(n)])/\gamma$.
      $b(n) = \mu(\Re[e(n)] - \Im[e(n)])/\gamma$.
      Add the new center $\boldsymbol{z}(n)$ to the list of centers, i.e., add $\boldsymbol{z}(n)$ to the list $\boldsymbol{Z}$, add $a(n)$ to the list $\boldsymbol{a}$, add $b(n)$ to the list $\boldsymbol{b}$.
   end for

$\nabla_{\boldsymbol{w}^*} \mathcal{L}_n(\boldsymbol{w}) = -e(n)\boldsymbol{\Phi}(\boldsymbol{z}(n))$. Following the same procedure, we conclude that the update rule becomes: $\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \mu e(n) \cdot \boldsymbol{\Phi}(\boldsymbol{z}(n))$, and assuming that $\boldsymbol{w}(0) = \boldsymbol{0}$, one concludes that:

$$\boldsymbol{w}(n) = \mu \sum_{k=1}^{n} e(k)\boldsymbol{\Phi}(\boldsymbol{z}(k)).$$

However, although this relation is different to equation (28), the filter output at iteration n, for this filter, turns out to be exactly the same as before:

$$\hat{d}(n) = \langle \boldsymbol{w}(n-1), \boldsymbol{\Phi}(\boldsymbol{z}(n)) \rangle_{\mathbb{H}}$$
$$= \mu \sum_{k=1}^{n-1} e(k)\langle \boldsymbol{\Phi}(\boldsymbol{z}(k)), \boldsymbol{\Phi}(\boldsymbol{z}(n)) \rangle_{\mathbb{H}},$$

which is in line with what we know for the standard complex LMS.

B. Complex KLMS with pure complex kernels - CKLMS2

As, in section VII-A, consider the sequence of examples $(\boldsymbol{z}(1), d(1)), (\boldsymbol{z}(2), d(2)), \ldots, (\boldsymbol{z}(N), d(N))$, where $d(n) \in \mathbb{C}$, $\boldsymbol{z}(n) \in V \subset \mathbb{C}^\nu$, $\boldsymbol{z}(n) = \boldsymbol{x}(n) + i\boldsymbol{y}(n)$, $\boldsymbol{x}(n), \boldsymbol{y}(n) \in \mathbb{R}^\nu$, for $n = 1, \ldots, N$. Consider also a complex reproducing kernel κ defined on $X \times X$, $X \subseteq \mathbb{C}^\nu$ and the respective complex RKHS $\mathbb{H}$. Each element $\boldsymbol{f} \in \mathbb{H}$ may be cast in the form $\boldsymbol{f} = u_{\boldsymbol{f}} + iv_{\boldsymbol{f}}$, $u_{\boldsymbol{f}}, v_{\boldsymbol{f}} \in \mathcal{H}$, where $\mathcal{H}$ is a real Hilbert space. We map the points $\boldsymbol{z}(n)$ to the complex RKHS $\mathbb{H}$ using the feature map $\tilde{\boldsymbol{\Phi}} : \mathbb{X} \to \mathbb{H} : \tilde{\boldsymbol{\Phi}}(\boldsymbol{z}) = \langle \cdot, \kappa(\cdot, \boldsymbol{z}) \rangle_{\mathbb{H}}$, for $n = 1, \ldots, N$. Estimating the filter output by $\hat{d}(n) = \langle \tilde{\boldsymbol{\Phi}}(\boldsymbol{z}(n)), \boldsymbol{w} \rangle_{\mathbb{H}}$, the objective of the complex Kernel LMS is to minimize $E[\mathcal{L}_n(\boldsymbol{w})]$, at each instance n. Once more, we apply the complex LMS to the transformed data, using the rules of Wirtinger's calculus to compute the gradient of $\mathcal{L}_n(\boldsymbol{w})$, i.e., $\nabla_{\boldsymbol{w}^*} \mathcal{L}_n(\boldsymbol{w}) = -e(n)^* \cdot \tilde{\boldsymbol{\Phi}}(\boldsymbol{z}(n))$. Therefore, the CKLMS2 update rule becomes $\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \mu e(n)^* \cdot \tilde{\boldsymbol{\Phi}}(\boldsymbol{z}(n))$, as expected, where $\boldsymbol{w}(n)$ denotes the estimate at iteration n.

Assuming that $\boldsymbol{w}(0) = \boldsymbol{0}$, the repeated application of the weight-update equation gives:

$$\boldsymbol{w}(n) = \sum_{k=1}^{n} e(k)^* \tilde{\boldsymbol{\Phi}}(\boldsymbol{z}(k)). \tag{30}$$

Thus, the filter output at iteration n becomes:

$$\hat{d}(n) = \langle \tilde{\boldsymbol{\Phi}}(\boldsymbol{z}(n)), \boldsymbol{w}(n-1) \rangle_{\mathbb{H}}$$
$$= \mu \sum_{k=1}^{n-1} e(k)\langle \tilde{\boldsymbol{\Phi}}(\boldsymbol{z}(n)), \tilde{\boldsymbol{\Phi}}(\boldsymbol{z}(k)) \rangle_{\mathbb{H}}$$
$$= \mu \sum_{k=1}^{n-1} e(k)\kappa(\boldsymbol{z}(k), \boldsymbol{z}(n)).$$

We should note, that the CKLMS2 algorithm may be equivalently derived, if one blindly applies the kernel trick on the complex LMS. However, such an approach conceals the mathematical framework that lies underneath, which is needed if one seeks a deeper understanding of the problem. The repeated application of the update equation of the CLMS yields:

$$\boldsymbol{w}(n) = \sum_{k=1}^{n} e(k)^* \boldsymbol{z}(k),$$

while the filter output at iteration n is given by:

$$\hat{d}(n) = \mu \sum_{k=1}^{n-1} e(k)\boldsymbol{z}(n)^H \boldsymbol{z}(k),$$

where the notation $\cdot^H$ denotes the Hermitian matrix. It is evident that the application of the kernel trick on these equations yields the same results.

Furthermore, note that, using the complex gaussian kernel, the algorithm is automatically normalized. The CKLMS2 algorithm is summarized in Algorithm 2.

Another formulation of the CKLMS2 algorithm may be derived if we estimate the filter output as $\hat{d}(n) = \langle \tilde{\boldsymbol{w}}(n-1), \boldsymbol{\Phi}(\boldsymbol{z}(n)) \rangle_{\mathbb{H}}$. Then the update rule becomes

$$\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \mu e(n) \cdot \tilde{\boldsymbol{\Phi}}(\boldsymbol{z}(n)).$$

Assuming that $\boldsymbol{w}(0) = \boldsymbol{0}$, the repeated application of the weight-update equation gives:

$$\boldsymbol{w}(n) = \sum_{k=1}^{n} e(k)\tilde{\boldsymbol{\Phi}}(\boldsymbol{z}(k)),$$

and the filter output at iteration n becomes:

$$\hat{d}(n) = \mu \sum_{k=1}^{n-1} e(k)\kappa(\boldsymbol{z}(n), \boldsymbol{z}(k)). \tag{31}$$

Note that the two formulations of the CKLMS2 are not identical, as it was the case for CKLMS. However, all the

simulated experiments that we performed, using the complex gaussian kernel, exhibited similar performance (in terms of signal to noise ratio - SNR).

Algorithm 2 Normalized Complex Kernel LMS2 (NCKLMS2)

INPUT: $(\boldsymbol{z}(1), d(1)), \ldots, (\boldsymbol{z}(N), d(N))$
OUTPUT: The expansion
$\boldsymbol{w} = \sum_{k=1}^{N} a(k)\kappa(\cdot, \boldsymbol{z}(k))$.

Initialization: Set $\boldsymbol{a} = \{\}$, $\boldsymbol{Z} = \{\}$ (i.e., $\boldsymbol{w} = \boldsymbol{0}$). Select the step parameter μ and the parameter σ of the complex gaussian kernel.

  for $n = 1 : N$ **do**
    Compute the filter output:

$$\hat{d}(n) = \sum_{k=1}^{n-1} a(k) \cdot \kappa(\boldsymbol{z}(k), \boldsymbol{z}(n)).$$

    Compute the error: $e(n) = d(n) - \hat{d}(n)$.
    $\gamma = \kappa(\boldsymbol{z}(n), \boldsymbol{z}(n))$.
    $a(n) = \mu e(n)/\gamma$.
    Add the new center $\boldsymbol{z}(n)$ to the list of centers, i.e., add $\boldsymbol{z}(n)$ to the list $\boldsymbol{Z}$, add $a(n)$ to the list $\boldsymbol{a}$.
  end for

C. Sparsification

The main drawback of any kernel based adaptive filtering algorithm is that a growing number of training points, $\boldsymbol{z}(n)$, is involved, as it is apparent from (28), (30) in the case of complex KLMS. Hence, increasing memory and computational resources are needed, as time evolves. Several strategies have been proposed to cope with this problem and to come up with sparse solutions. In this paper, we employ the well known *novelty criterion* [14], [37]. In novelty criterion online sparsification, a dictionary of points, C, is formed and updated appropriately. Whenever a new data pair $(\boldsymbol{\Phi}(\boldsymbol{z}_n), d_n)$ is considered, a decision is immediately made of whether to add the new center, $\boldsymbol{\Phi}(\boldsymbol{z}(n))$, to the dictionary of centers $\mathcal{C}$. The decision is reached following two simple rules. First, the distance of the new center, $\boldsymbol{\Phi}(\boldsymbol{z}(n))$, from the current dictionary is evaluated: $dis = \min_{\boldsymbol{c}_k \in \mathcal{C}}\{\|\boldsymbol{\Phi}(\boldsymbol{z}(n)) - \boldsymbol{c}_k\|_{\mathbb{H}}\}$. If this distance is smaller than a given threshold δ_1 (i.e., the new center is close to the existing dictionary), then the center is not added to $\mathcal{C}$. Otherwise, we compute the prediction error $e_n = d_n - \hat{d}_n$. If $|e_n|$ is smaller than a predefined threshold δ_2, then the new center is discarded. Only if $|e_n| \geq \delta_2$ the new center $\boldsymbol{\Phi}(\boldsymbol{z}(n))$ is added to the dictionary.

An alternative method has been considered in [4], which results in an exponential forgetting mechanism of past data. In [6], [38], the sliding window rationale has been considered. In all the implementations of CKLMS that are presented in this paper the novelty criterion was adopted.

VIII. EXPERIMENTS

The performances of CKLMS1 and CKLMS2 have been tested in the context of: a) a nonlinear channel equalization
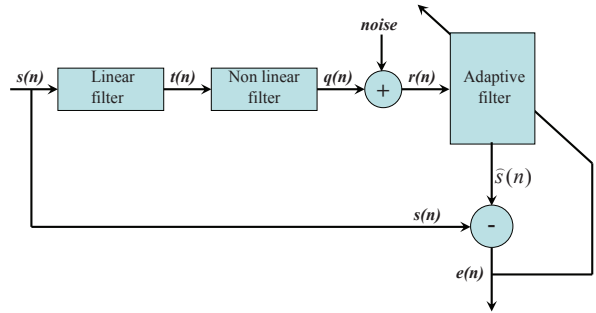

Fig. 1. The equalization task.

task (see figure 1) and b) a nonlinear channel identification task.

A. Channel Equalization

For the first case, 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) is comprised by 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 [1]. 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

$$s(n) = 0.70 \left(\sqrt{1 - \rho^2} X(n) + i\rho Y(n) \right), \qquad (32)$$

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 [18]. Note that the issue of circularity is very important in complex adaptive filtering. 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 [17]. Loosely speaking, non circularity adds some form of nonlinearity to the signal. It can be proved that widely linear estimation (i.e., linear estimation in both z and z^*) outperforms standard linear estimation for general (i.e., circular or non-circular) complex signals. For circular signals, the two models lead to identical results [16], [39].

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 close as possible. To this end, we apply the NCKLMS1 and the NCKLMS2 algorithms 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 (32) considering both the circular and the non-circular cases. The results are compared with the NCLMS and the WL-NCLMS (i.e., widely linear NCLMS) algorithms and with two adaptive nonlinear algorithms: a) the CNGD algorithm, which is thoroughly described in [17] and a Multi Layer Perceptron (MLP) with 50 nodes in the hidden layer (proposed in [18]). In both cases, the complex tanh activation function was employed. Note that the WL-NCLMS has been recently used as an alternative to the CLMS, in an attempt to cope with non circularity as well as with soft nonlinearities. In all algorithms, the step update parameter, μ, is tuned for the best possible results (in terms of the steady-state error rate). For the case of the MLP, the design was also tuned so that the best possible results were obtained. Time delay D was also set for optimality. Figure 2, shows the learning curves of the NCKLMS1 using the real Gaussian kernel $\kappa(\boldsymbol{x}, \boldsymbol{y}) = \exp(-\|\boldsymbol{x} - \boldsymbol{y}\|^2/\sigma^2)$ (with $\sigma = 5$) and the NCKLMS2 using the complex Gaussian kernel $\kappa_{\sigma,\mathbb{C}^d}(\boldsymbol{z}, \boldsymbol{w}) := \exp\left(-\frac{\sum_{i=1}^d (z_i - w_i^*)^2}{\sigma^2}\right)$ (with $\sigma = 5$), together with those obtained from the NCLMS and the WL-NCLMS algorithms. Figure 3 shows the learning curves of the NCKLMS1 and NCKLMS2 versus the CNGD and the L-50-1 MLP. Finally, figure 4 compares the learning curves of NCKLMS1 versus a split channel approach, that treats the complex signal as two real ones using the KLMS.

The novelty criterion was used for the sparsification of the NCKLMS1 with $\delta_1 = 0.15$ and $\delta_2 = 0.2$ and of the NCKLMS2 with $\delta_1 = 0.1$ and $\delta_2 = 0.2$. In both examples, NCKLMS1 considerably outperforms the linear, widely linear (i.e., NCLMS and WL-NCLMS) and nonlinear (CNGD and MLP) algorithms (see figures 2, 3). The NCKLMS2 also exhibits improved performance compared to the linear, widely linear and nonlinear algorithms. However, in both cases, this enhanced behavior comes at a price in computational complexity, since the NCKLMS requires the evaluation of the kernel function. In terms of the required computer time, the complexity of CKLMS1 and CKLMS2 is of the same order as the complexity of the MLP. Comparing the NCKLMS1 and the NCKLMS2, the experiments show that the results differ, with the former one leading to an improved performance. Finally, figure 4 illustrates that the split channel approach performs poorly compared to the NCKLMS1, especially in the circular case, as it cannot capture the correlation between the two real channels.
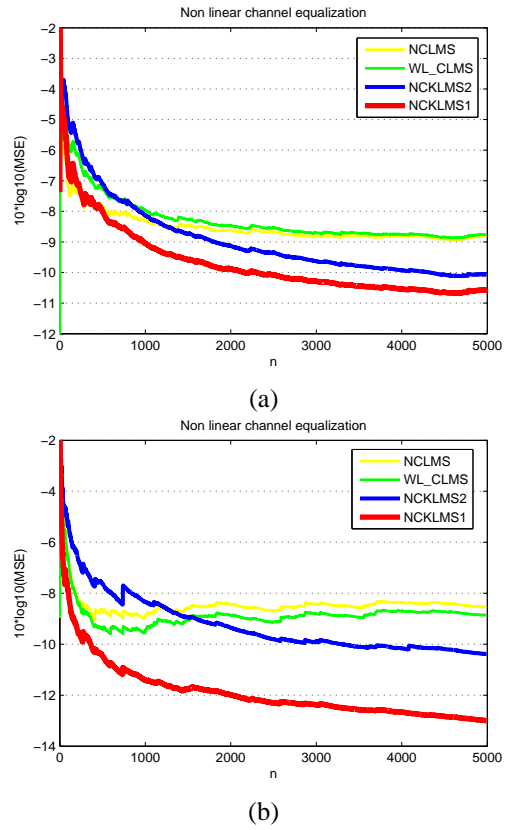

(a)

(b)

Fig. 2. Learning curves for KCLMS1 ($\mu = 1/2$), KCLMS2, ($\mu = 1/4$), CLMS ($\mu = 1/16$) and WL-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 ($\rho = 0.1$).

B. Channel Identification

The nonlinear channel that was considered (see [18]) consists of a linear filter:

$$t(n) = \sum_{k=1}^{5} h(k) \cdot s(n - k + 1),$$

where

$$h(k) = 0.432 \left(1 + \cos\left(\frac{2\pi(k-3)}{5}\right) - \left(1 + \cos\frac{2\pi(k-3)}{10}\right) i\right),$$

for $k = 1, \ldots, 5$, and the nonlinear component:

$$x(n) = t(n) + (0.15 - 0.1i)t^2(n).$$

Similar to the equalization case, the input signal that was fed to the channel had the form (32). Experiments were conducted on a set of 10000 samples of the input signal (32), corrupted by white gaussian noise, considering both the circular and the non-circular case. The level of the noise was set to 18dB. Figure 5, shows the learning curves of the NCKLMS1 and the NCKLMS2 together with those obtained from the CNGD and the L-50-1 MLP. In this example, also, NCKLMS1 considerably outperforms both the CNGD and the L-50-1 MLP. The NCKLMS2 although performs better than MLP, CNGD, its performance is inferior to NCKLMS1.

IX. CONCLUSIONS

A new framework for kernel adaptive filtering for complex signal processing has been developed. The proposed methodology, besides providing a skeleton for working with pure complex kernels, allows
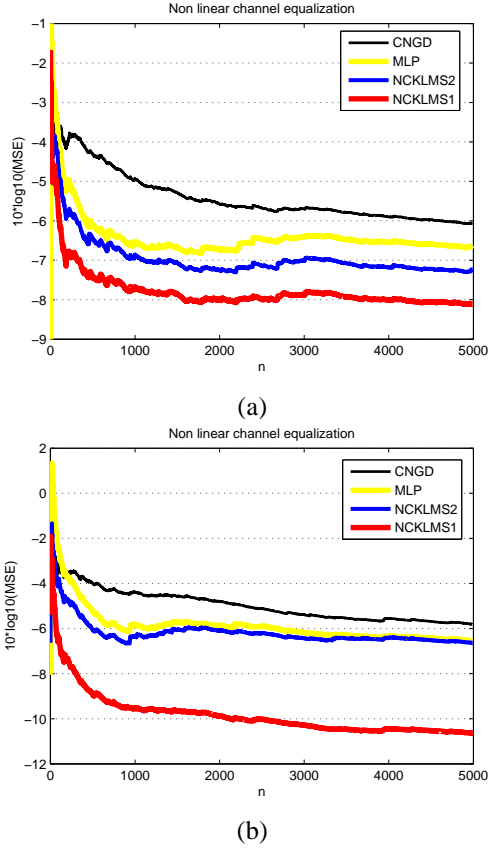
(a)

(b)

Fig. 3. Learning curves for KCLMS1 ($\mu = 1/2$), KCLMS2, ($\mu = 1/4$), CNGD and L-50-1 MLP (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 ($\rho = 0.1$).
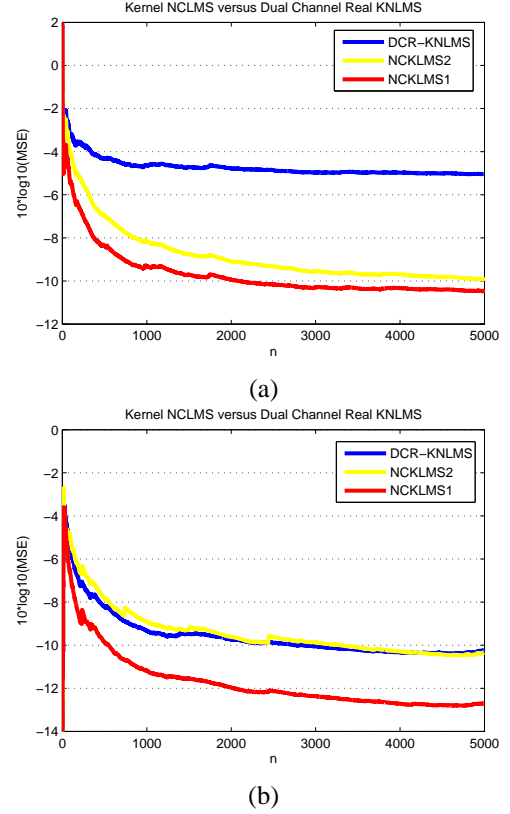

(a)

(b)

Fig. 4. Learning curves for KCLMS1 ($\mu = 1/2$) and Dual Channel Real KLMS ($\mu = 1/2$) for the soft nonlinear channel equalization problem, for (a) the circular input case, (b) the non-circular input case ($\rho = 0.1$).

for the construction of complex RKHSs from real ones, through a technique called complexification of RKHSs. Such an approach provides the advantage of working with popular and well understood real kernels in the complex domain. It has to be pointed out, that our method is a general one and can be used on any type of real and/or complex kernels that have or can be developed. To the best of our knowledge, this is the first time that a methodology for complex adaptive processing in RKHSs is proposed. Wirtinger's calculus has been extended to cope with the problem of differentiation in the involved (infinite) dimensional Hilbert spaces. The derived rules and properties of the extended Wirtinger's calculus on complex RKHS turn out to be similar in structure to the special case of finite dimensional complex spaces. The proposed framework was applied on the complex LMS and two realizations for the complex Kernel LMS algorithm were developed. Experiments, which were performed on both the equalization and the identification problem of a nonlinear channel, for both circular and non-circular input data, showed a significant decrease in the steady state mean square error, compared with other known linear, widely linear and nonlinear techniques, while retaining a fast convergence.

APPENDIX A
PROOF OF PROPOSITION VI.2

We start with a lemma that will be used to prove the claim.

Lemma A.1. *Consider the Hilbert space $\mathbb{H}$ and $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{H}$. The limit*

$$\lim_{\|\boldsymbol{h}\|_{\mathbb{H}} \to 0} \frac{\langle \boldsymbol{h}^*, \boldsymbol{a} \rangle_{\mathbb{H}} - \langle \boldsymbol{h}, \boldsymbol{b} \rangle_{\mathbb{H}}}{\|\boldsymbol{h}\|_{\mathbb{H}}} = 0, \qquad (33)$$

if and only if $\boldsymbol{a} = \boldsymbol{b} = 0$.

Consider the first order Taylor expansions of T_r and T_i at $\boldsymbol{c} = c_1 + ic_2 = (c_1, c_2)$:

$$\begin{aligned}
T_r(\boldsymbol{c} + \boldsymbol{h}) &= T_r(\boldsymbol{c}) + \langle h_1, \nabla_u T_r(\boldsymbol{c}) \rangle_{\mathcal{H}} \\
&\quad + \langle h_2, \nabla_v T_r(\boldsymbol{c}) \rangle_{\mathcal{H}} + o(\|\boldsymbol{h}\|_{\mathcal{H}^2}), \\
T_i(\boldsymbol{c} + \boldsymbol{h}) &= T_i(\boldsymbol{c}) + \langle h_1, \nabla_u T_i(\boldsymbol{c}) \rangle_{\mathcal{H}} \\
&\quad + \langle h_2, \nabla_v T_i(\boldsymbol{c}) \rangle_{\mathcal{H}} + o(\|\boldsymbol{h}\|_{\mathcal{H}^2}).
\end{aligned}$$

Multiplying the second relation with i and adding it to the first one, we take:

$$\begin{aligned}
\boldsymbol{T}(\boldsymbol{c} + \boldsymbol{h}) &= \boldsymbol{T}(\boldsymbol{c}) + \langle h_1, \nabla_u T_r(\boldsymbol{c}) - i\nabla_u T_i(\boldsymbol{c}) \rangle_{\mathbb{H}} \\
&\quad + \langle h_2, \nabla_v T_r(\boldsymbol{c}) - i\nabla_v T_i(\boldsymbol{c}) \rangle_{\mathbb{H}} + o(\|\boldsymbol{h}\|_{\mathbb{H}}).
\end{aligned}$$

To simplify the notation we may define

$$\begin{aligned}
\nabla_u \boldsymbol{T}(\boldsymbol{c}) &= \nabla_u T_r(\boldsymbol{c}) + i\nabla_u T_i(\boldsymbol{c}) \\
\nabla_v \boldsymbol{T}(\boldsymbol{c}) &= \nabla_v T_r(\boldsymbol{c}) + i\nabla_v T_i(\boldsymbol{c})
\end{aligned}$$

and obtain:

$$\begin{aligned}
\boldsymbol{T}(\boldsymbol{c} + \boldsymbol{h}) &= \boldsymbol{T}(\boldsymbol{c}) + \langle h_1, (\nabla_u \boldsymbol{T}(\boldsymbol{c}))^* \rangle_{\mathbb{H}} \\
&\quad + \langle h_2, (\nabla_v \boldsymbol{T}(\boldsymbol{c}))^* \rangle_{\mathbb{H}} + o(\|\boldsymbol{h}\|_{\mathcal{H}^2}).
\end{aligned}$$

Next, we substitute h_1 and h_2 using the relations $h_1 = \frac{\boldsymbol{h} + \boldsymbol{h}^*}{2}$ and $h_2 = \frac{\boldsymbol{h} - \boldsymbol{h}^*}{2i}$ and use the sesquilinear property of the inner product of $\mathbb{H}$:

$$\begin{aligned}
\boldsymbol{T}(\boldsymbol{c} + \boldsymbol{h}) &= \boldsymbol{T}(\boldsymbol{c}) + \frac{1}{2} \langle \boldsymbol{h}, (\nabla_u \boldsymbol{T}(\boldsymbol{c}) - i\nabla_v \boldsymbol{T}(\boldsymbol{c}))^* \rangle_{\mathbb{H}} \\
&\quad + \frac{1}{2} \langle \boldsymbol{h}^*, (\nabla_u \boldsymbol{T}(\boldsymbol{c}) + i\nabla_v \boldsymbol{T}(\boldsymbol{c}))^* \rangle_{\mathbb{H}} + o(\|\boldsymbol{h}\|_{\mathbb{H}}).
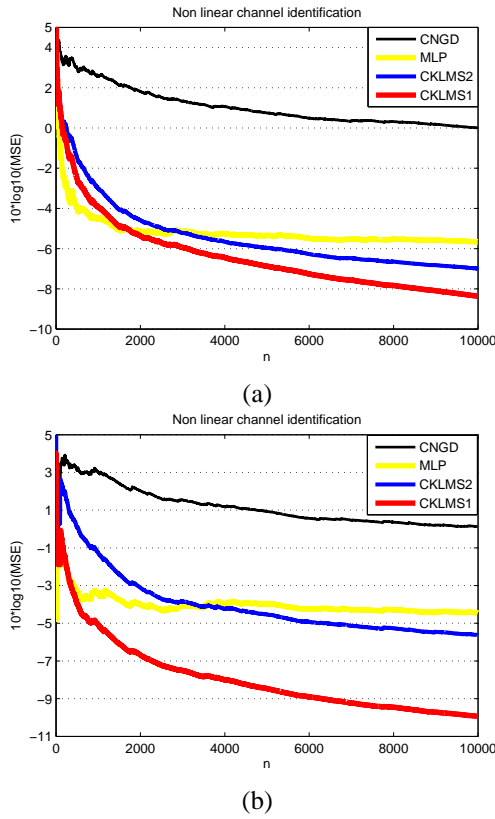\end{aligned}$$

(a)

(b)

Fig. 5.   Learning curves for KCLMS1 ($\mu = 1/2$), KCLMS2, ($\mu = 1/4$), CNGD and L-50-1 MLP (filter length $L = 5$) for the nonlinear channel identification problem, for (a) the circular input case, (b) the non-circular input case ($\rho = 0.1$).

It has already been shown that equation (21) is essential for the development of Wirtinger's calculus. To complete the proof of the proposition we compute the fraction that appears in the definition of the complex Fréchet derivative:

$$\frac{\boldsymbol{T}(\boldsymbol{c}+\boldsymbol{h}) - \boldsymbol{T}(\boldsymbol{c}) - \langle \boldsymbol{h}, \boldsymbol{w}\rangle_{\mathbb{H}}}{\|\boldsymbol{h}\|_{\mathbb{H}}} =$$

$$\left(\frac{1}{2}\langle \boldsymbol{h}, (\nabla_u \boldsymbol{T}(\boldsymbol{c}) - i\nabla_v \boldsymbol{T}(\boldsymbol{c}))^* \rangle_{\mathbb{H}} \right.$$

$$\left. + \frac{1}{2}\langle \boldsymbol{h}^*, (\nabla_u \boldsymbol{T}(\boldsymbol{c}) + i\nabla_v \boldsymbol{T}(\boldsymbol{c}))^* \rangle_{\mathbb{H}} - \langle \boldsymbol{h}, \boldsymbol{w}\rangle_{\mathbb{H}} \right) \Big/ \|\boldsymbol{h}\|_{\mathbb{H}}$$

$$+ \frac{o(\|\boldsymbol{h}\|_{\mathbb{H}})}{\|\boldsymbol{h}\|_{\mathbb{H}}}.$$

Recall that, since $o(\|\boldsymbol{h}\|_{\mathbb{H}})/\|\boldsymbol{h}\|_{\mathbb{H}} \to 0$ as $\|\boldsymbol{h}\|_{\mathbb{H}} \to 0$, for this limit to exist and vanish, it is necessary that $\nabla_u \boldsymbol{T}(\boldsymbol{c}) + i\nabla_v \boldsymbol{T}(\boldsymbol{c}) = 0$ and $\boldsymbol{w}^* = \nabla_u \boldsymbol{T}(\boldsymbol{c}) - i\nabla_v \boldsymbol{T}(\boldsymbol{c})$ (see lemma A.1). However, according to our definition,

$$\nabla_u \boldsymbol{T}(\boldsymbol{c}) + i\nabla_v \boldsymbol{T}(\boldsymbol{c}) = (\nabla_u T_r(\boldsymbol{c}) - \nabla_v T_i(\boldsymbol{c}))$$
$$+ i\left(\nabla_u T_i(\boldsymbol{c}) + \nabla_v T_r(\boldsymbol{c})\right).$$

Thus, $\boldsymbol{T}$ is differentiable in the Fréchet complex sense, iff the Cauchy-Riemann conditions hold. Moreover, in this case:

$$\nabla \boldsymbol{T}(\boldsymbol{c}) = \nabla_u T_r(\boldsymbol{c}) + i\nabla_u T_i(\boldsymbol{c})$$
$$= \nabla_v T_i(\boldsymbol{c}) - i\nabla_v T_r(\boldsymbol{c}).$$

APPENDIX B
PROPERTIES OF WIRTINGER'S DERIVATIVES ON COMPLEX HILBERT SPACES

Below we give a complete list of the main properties of the extended Wirtinger's Calculus in complex Hilbert spaces. A rigorous and detailed presentation of the theory, as well as the proofs of all these properties can be found in [35].

1) If $\boldsymbol{T}(\boldsymbol{f})$ is $\boldsymbol{f}$-holomorphic at $\boldsymbol{c}$ (i.e., it has a Taylor series expansion with respect to $\boldsymbol{f}$ around $\boldsymbol{c}$), then its Fréchet W-derivative at $\boldsymbol{c}$ degenerates to the standard Fréchet complex derivative and its Fréchet CW-derivative vanishes, i.e., $\nabla_{\boldsymbol{f}^*}\boldsymbol{T}(\boldsymbol{c}) = \boldsymbol{0}$.

2) If $\boldsymbol{T}(\boldsymbol{f})$ is $\boldsymbol{f}^*$-holomorphic at $\boldsymbol{c}$ (i.e., it has a Taylor series expansion with respect to $\boldsymbol{f}^*$ around $\boldsymbol{c}$), then $\nabla_{\boldsymbol{f}}\boldsymbol{T}(\boldsymbol{c}) = \boldsymbol{0}$.

3) $(\nabla_{\boldsymbol{f}}\boldsymbol{T}(\boldsymbol{c}))^* = \nabla_{\boldsymbol{f}^*}\boldsymbol{T}^*(\boldsymbol{c})$.

4) $(\nabla_{\boldsymbol{f}^*}\boldsymbol{T}(\boldsymbol{c}))^* = \nabla_{\boldsymbol{f}}\boldsymbol{T}^*(\boldsymbol{c})$.

5) If $\boldsymbol{T}$ is real valued, then $(\nabla_{\boldsymbol{f}}\boldsymbol{T}(\boldsymbol{c}))^* = \nabla_{\boldsymbol{f}^*}\boldsymbol{T}(\boldsymbol{c})$.

6) The first order Taylor expansion around $\boldsymbol{f} \in \mathbb{H}$ is given by

$$\boldsymbol{T}(\boldsymbol{f}+\boldsymbol{h}) = \boldsymbol{T}(\boldsymbol{f}) + \langle \boldsymbol{h}, (\nabla_{\boldsymbol{f}}\boldsymbol{T}(\boldsymbol{f}))^* \rangle_{\mathbb{H}} + \langle \boldsymbol{h}^*, (\nabla_{\boldsymbol{f}^*}\boldsymbol{T}(\boldsymbol{f}))^* \rangle_{\mathbb{H}}.$$

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

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

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

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

11) Linearity: If $\boldsymbol{T}, \boldsymbol{S} : \mathbb{H} \to \mathbb{C}$ are Fréchet differentiable in the real sense at $\boldsymbol{c} \in \mathbb{H}$ and $\alpha, \beta \in \mathbb{C}$, then

$$\nabla_{\boldsymbol{f}}(\alpha\boldsymbol{T} + \beta\boldsymbol{S})(\boldsymbol{c}) = \alpha\nabla_{\boldsymbol{f}}\boldsymbol{T}(\boldsymbol{c}) + \beta\nabla_{\boldsymbol{f}}\boldsymbol{S}(\boldsymbol{c})$$
$$\nabla_{\boldsymbol{f}^*}(\alpha\boldsymbol{T} + \beta\boldsymbol{S})(\boldsymbol{c}) = \alpha\nabla_{\boldsymbol{f}^*}\boldsymbol{T}(\boldsymbol{c}) + \beta\nabla_{\boldsymbol{f}^*}\boldsymbol{S}(\boldsymbol{c}).$$

12) Product Rule: If $\boldsymbol{T}, \boldsymbol{S} : \mathbb{H} \to \mathbb{C}$ are Fréchet differentiable in the real sense at $\boldsymbol{c} \in \mathbb{H}$, then:

$$\nabla_{\boldsymbol{f}}(\boldsymbol{T} \cdot \boldsymbol{S})(\boldsymbol{c}) = \nabla_{\boldsymbol{f}}\boldsymbol{T}(\boldsymbol{c})\boldsymbol{S}(\boldsymbol{c}) + \boldsymbol{T}(\boldsymbol{c})\nabla_{\boldsymbol{f}}\boldsymbol{S}(\boldsymbol{c}),$$
$$\nabla_{\boldsymbol{f}^*}(\boldsymbol{T} \cdot \boldsymbol{S})(\boldsymbol{c}) = \nabla_{\boldsymbol{f}^*}\boldsymbol{T}(\boldsymbol{c})\boldsymbol{S}(\boldsymbol{c}) + \boldsymbol{T}(\boldsymbol{c})\nabla_{\boldsymbol{f}^*}\boldsymbol{S}(\boldsymbol{c}).$$

13) Division Rule: If $\boldsymbol{T}, \boldsymbol{S} : \mathbb{H} \to \mathbb{C}$ are Fréchet differentiable in the real sense at $\boldsymbol{c} \in \mathbb{H}$ and $\boldsymbol{S}(\boldsymbol{c}) \neq 0$, then:

$$\nabla_{\boldsymbol{f}}\left(\frac{\boldsymbol{T}}{\boldsymbol{S}}\right)(\boldsymbol{c}) = \frac{\nabla_{\boldsymbol{f}}\boldsymbol{T}(\boldsymbol{c})\boldsymbol{S}(\boldsymbol{c}) - \boldsymbol{T}(\boldsymbol{c})\nabla_{\boldsymbol{f}}\boldsymbol{S}(\boldsymbol{c})}{\boldsymbol{S}^2(\boldsymbol{c})},$$
$$\nabla_{\boldsymbol{f}^*}\left(\frac{\boldsymbol{T}}{\boldsymbol{S}}\right)(\boldsymbol{c}) = \frac{\nabla_{\boldsymbol{f}^*}\boldsymbol{T}(\boldsymbol{c})\boldsymbol{S}(\boldsymbol{c}) - \boldsymbol{T}(\boldsymbol{c})\nabla_{\boldsymbol{f}^*}\boldsymbol{S}(\boldsymbol{c})}{\boldsymbol{S}^2(\boldsymbol{c})}.$$

14) Chain Rule: If $\boldsymbol{T} : \mathbb{H} \to \mathbb{C}$ is Fréchet differentiable at $\boldsymbol{c} \in \mathbb{H}$, $\boldsymbol{S} : \mathbb{C} \to \mathbb{C}$ is differentiable in the real sense at $\boldsymbol{T}(\boldsymbol{c}) \in \mathbb{C}$, then:

$$\nabla_{\boldsymbol{f}}\boldsymbol{S} \circ \boldsymbol{T}(\boldsymbol{c}) = \frac{\partial \boldsymbol{S}}{\partial z}(\boldsymbol{T}(\boldsymbol{c}))\nabla_{\boldsymbol{f}}\boldsymbol{T}(\boldsymbol{c})$$
$$+ \frac{\partial \boldsymbol{S}}{\partial z^*}(\boldsymbol{T}(\boldsymbol{c}))\nabla_{\boldsymbol{f}}(\boldsymbol{T}^*)(\boldsymbol{c}),$$
$$\nabla_{\boldsymbol{f}^*}\boldsymbol{S} \circ \boldsymbol{T}(\boldsymbol{c}) = \frac{\partial \boldsymbol{S}}{\partial z}(\boldsymbol{T}(\boldsymbol{c}))\nabla_{\boldsymbol{f}^*}\boldsymbol{T}(\boldsymbol{c})$$
$$+ \frac{\partial \boldsymbol{S}}{\partial z^*}(\boldsymbol{T}(\boldsymbol{c}))\nabla_{\boldsymbol{f}^*}(\boldsymbol{T}^*)(\boldsymbol{c}).$$

The proofs of properties 1 and 2 are rather obvious. Here, we give the proofs of properties 3, 7 and 11, which have been used to derive the main results of this paper.

Proof of property 3: The existence of $\nabla_{\boldsymbol{f}}\boldsymbol{T}(\boldsymbol{c})$ and $\nabla_{\boldsymbol{f}^*}\boldsymbol{T}(\boldsymbol{c})$ is guaranteed by the Fréchet differentiability of $\boldsymbol{T}$ at $\boldsymbol{c}$ (in the real

sense). To take the result, observe that:

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

Property 4 can be proved similarly.

Proof of property 7: Considering the definition of Fréchet complex derivative (see equation 17), we observe that:

$$\begin{aligned}
\boldsymbol{T}(\boldsymbol{c}+\boldsymbol{h}) - \boldsymbol{T}(\boldsymbol{c}) &- \langle\boldsymbol{h},\boldsymbol{g}\rangle_{\mathbb{H}} = \\
\langle\boldsymbol{c}+\boldsymbol{h},\boldsymbol{w}\rangle_{\mathbb{H}} &- \langle\boldsymbol{c},\boldsymbol{w}\rangle_{\mathbb{H}} - \langle\boldsymbol{h},\boldsymbol{g}\rangle_{\mathbb{H}} \\
&= \langle\boldsymbol{h},\boldsymbol{w}\rangle_{\mathbb{H}} - \langle\boldsymbol{h},\boldsymbol{g}\rangle_{\mathbb{H}}.
\end{aligned}$$

Thus, $\boldsymbol{T}$ is Fréchet complex differentiable at $\boldsymbol{c}$, with $\nabla\boldsymbol{T}(\boldsymbol{c}) = \boldsymbol{w}^*$ and from property 1, $\nabla_{\boldsymbol{f}^*}(\boldsymbol{c}) = \boldsymbol{0}$ and $\nabla_{\boldsymbol{f}}(\boldsymbol{c}) = \boldsymbol{w}$.

Proof of property 11: Let $\boldsymbol{T}(\boldsymbol{f}) = T_r(u_{\boldsymbol{f}}, v_{\boldsymbol{f}}) + iT_i(u_{\boldsymbol{f}}, v_{\boldsymbol{f}})$, $\boldsymbol{S}(\boldsymbol{f}) = \boldsymbol{S}(u_{\boldsymbol{f}} + iv_{\boldsymbol{f}}) = S_r(u_{\boldsymbol{f}}, v_{\boldsymbol{f}}) + iS_i(u_{\boldsymbol{f}}, v_{\boldsymbol{f}})$ be two complex functions and $\alpha, \beta \in \mathbb{C}$, such that $\alpha = \alpha_1 + i\alpha_2$, $\beta = \beta_1 + i\beta_2$. Then $\boldsymbol{R}(\boldsymbol{f}) = \alpha\boldsymbol{T}(\boldsymbol{f}) + \beta\boldsymbol{S}(\boldsymbol{f})$ and the Fréchet W-derivative of $\boldsymbol{R}$ will be given by:

$$\nabla_{\boldsymbol{f}}\boldsymbol{R}(\boldsymbol{c}) = \frac{1}{2}\left(\nabla_u R_r(\boldsymbol{c}) + \nabla_v R_i(\boldsymbol{c})\right) + \frac{i}{2}\left(\nabla_u R_i(\boldsymbol{c}) - \nabla_v R_r(\boldsymbol{c})\right).$$

Applying the linearity property of the ordinary Fréchet derivative, after some algebra we take the result. For the second part, in view of properties 3, 4 and the linearity property of the Fréchet W-derivative, the Fréchet CW-derivative of $\boldsymbol{R}$ at $\boldsymbol{c}$ will be given by:

$$\begin{aligned}
\nabla_{\boldsymbol{f}^*}\boldsymbol{R}(\boldsymbol{c}) =& \nabla_{\boldsymbol{f}^*}(\alpha\boldsymbol{T} + \beta\boldsymbol{S})(\boldsymbol{c}) \\
=& \left(\nabla_{\boldsymbol{f}}(\alpha\boldsymbol{T} + \beta\boldsymbol{S})^*(\boldsymbol{c})\right)^* \\
=& \left(\nabla_{\boldsymbol{f}}(\alpha^*\boldsymbol{T}^* + \beta^*\boldsymbol{S}^*)(\boldsymbol{c})\right)^* \\
=& \left(\alpha^*\nabla_{\boldsymbol{f}}\boldsymbol{T}^*(\boldsymbol{c}) + \beta^*\nabla_{\boldsymbol{f}}\boldsymbol{S}^*(\boldsymbol{c})\right)^* \\
=& \alpha\left(\nabla_{\boldsymbol{f}}\boldsymbol{T}^*(\boldsymbol{c})\right)^* + \beta\left(\nabla_{\boldsymbol{f}}\boldsymbol{S}^*(\boldsymbol{c})\right)^* \\
=& \alpha\nabla_{\boldsymbol{f}^*}\boldsymbol{T}(\boldsymbol{c}) + \beta\nabla_{\boldsymbol{f}^*}\boldsymbol{S}(\boldsymbol{c}),
\end{aligned}$$

which completes the proof.

REFERENCES

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

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

[3] Y. Engel, S. Mannor, and R. Meir, "The kernel recursive least-squares algorithm," *IEEE Trans. Sign. Proc.*, vol. 52, no. 8, 2004.

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

[5] ——, "Adaptive constrained learning in reproducing kernel hilbert spaces: The robust beamforming case," *IEEE Trans. Signal Process.*, vol. 57, no. 12, pp. 4744–4764, 2009.

[6] K. Slavakis and S. Theodoridis, "Sliding window generalized kernel affine projection algorithm using projection mappings," *Eurasip Journal on Advances in Signal Processing*, vol. art. no. 735351, 2008.

[7] B. Schölkopf and A. J. Smola, *Learning with Kernels*. MIT Press, 2002.

[8] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, 4th edition*. Academic Press, 2009.

[9] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[10] K. Kim, M. O. Franz, and B. Scholkopf, "Iterative kernel principal component analysis for image modeling," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 9, pp. 1351–1366, 2005.

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

[12] B. Schölkopf, A. J. Smola, and K. R. Muller, "Kernel principal component analysis," *Lecture notes in computer science*, vol. 1327, pp. 583–588, 1997.

[13] M. Filippone, F. Camastra, F. Masulli, and S. Rovetta, "A survey of kernel and spectral methods for clustering. pattern recognition," *Pattern Recognition*, vol. 41, no. 1, pp. 176–190, 2008.

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

[15] W. Wirtinger, "Zur formalen theorie der functionen von mehr complexen veränderlichen," *Math. Ann.*, vol. 97, pp. 357–375, 1927.

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

[17] D. Mandic and V. S. L. Goh, *Complex Valued Nonlinear Adaptive Filters*. Wiley, 2009.

[18] T. Adali and H. Li, *Complex-valued adaptive signal processing*, ser. Adaptive Signal Processing: Next Generation Solutions, T. Adali and S. Haykin, editors. Hoboken, NJ, Wiley, 2010.

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

[20] M. Novey and T. Adali, "On extending the complex fast ICA algorithm to noncircular sources," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 2148–2154, 2008.

[21] H. Li and T. Adali, "Complex-valued adaptive signal processing using nonlinear functions," *Journal on Advances in Signal Processing, Special issue on Emerging Machine Learning Techniques in Signal Processing*, pp. Article ID 765 615, 9 pages, 2008.

[22] H. L. T. Adali and R. Aloysius, "On properties of the widely linear mse filter and its lms implementation," *Proc. Conf. on Info. Sciences and Systems (CISS), Baltimore, MD, March 2009*.

[23] T. Adali and S. Haykin, *Adaptive Signal Processing: Next Generation Solutions*. Wiley-IEEE Press, 2010.

[24] S. Saitoh, *Integral Transforms, Reproducing Kernels and their applications*. Longman Scientific & Technical, Harlow, 1997.

[25] V. I. Paulsen, "An introduction to the theory of reproducing kernel hilbert spaces," http://www.math.uh.edu/~vern/rkhs.pdf.

[26] J. Mercer, "Functions of positive and negative type and their connection with the theory of integral equations," *Phil. Trans. Roy. Soc. Ser. A*, vol. 209, pp. 415–446, 1909.

[27] N. Aronszajn, "Theory of reproducing kernels," *Transactions of the American Mathematical Society*, vol. 68, no. 3, pp. 337–404, 1950.

[28] I. Steinwart, D. Hush, and C. Scovel, "An explicit description of the reproducing kernel hilbert spaces of gaussian rbf kernels," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4635–4643, 2006.

[29] D. Luenberger, *Linear and Nonlinear Programming*. Addison-Wesley, 1984.

[30] Liusternik and Sobolev, *Elements of Functional Analysis*. Frederick Ungar Publishing Co, 1961.

[31] A. V. Balakrishnan, *Applied Functional Analysis*. Springer, 1976.

[32] D. H. Brandwood, "A complex gradient operator and its application in adaptive array theory," *IEE proc. H (Microwaves, optics and Antennas)*, vol. 130, no. 1, pp. 11–16, 1983.

[33] A. V. de Bos, "Complex gradient and hessian," *IEE proc. Visual image signal processing*, vol. 141, no. 6, pp. 380–382, 1994.

[34] K. Kreutz-Delgado, "The complex gradient operator and the $\mathbb{CR}$-calculus," http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.86.6515&rep=rep1&type=pdf.

[35] P. Bouboulis, "Wirtinger's calculus in general hilbert spaces," http://arxiv.org/abs/1005.5170.

[36] S. B. Chae, *Holomorphy and Calculus in Normed Spaces*. Marcel Dekker, 1985.

[37] J. Platt, "A resourse allocating network for function interpolation," *Newral Computation*, vol. 3, no. 2, pp. 213–225, 1991.

[38] S. van Vaerenbergh, J. Vía, and I. Santamaría, "A sliding-window kernel RLS algorithm and its application to nonlinear channel identification," in *Proceedings of ICASSP*, vol. V. Toulouse, France: IEEE, 2006, pp. 789–792.

[39] B. Picinbono, "On circularity," *IEEE Trans. Signal Process.*, vol. 42, no. 12, pp. 3473 – 3482, 1994.

PLACE
PHOTO
HERE

Pantelis Bouboulis (M' 10) received the M.Sc. and Ph.D. degrees in informatics and telecommunications from the National and Kapodistrian University of Athens, Greece, in 2002 and 2006, respectively. From 2007 till 2008, he served as an Assistant Professor in the Department of Informatics and Telecommunications, University of Athens. His current research interests lie in the areas of machine learning, fractals, wavelets and image processing. He is a member of AMS and IEEE.

PLACE
PHOTO
HERE

Sergios Theodoridis (F08) is currently Professor of signal processing and communications in the Department of Informatics and Telecommunications, University of Athens, Athens, Greece. His research interests lie in the areas of adaptive algorithms and communications, machine learning and pattern recognition, and signal processing for audio processing and retrieval. He is the co-editor of the book Efficient Algorithms for Signal Processing and System Identification (Prentice-Hall, 1993), coauthor of the best selling book Pattern Recognition (Academic, 4th ed., 2008), and the coauthor of three books in Greek, two of them for the Greek Open University. He is currently an Associate Editor of the IEEE TRANSACTIONS ON NEURAL NETWORKS, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II, and a member of the editorial board of the EURASIP Wireless Communications and Networking. He has served in the past as an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE Signal Processing Magazine, the EURASIP Journal on Signal Processing, and the EURASIP Journal on Advances on Signal Processing. He was the general chairman of EUSIPCO 1998, the Technical Program cochair for ISCAS 2006, and cochairman of ICIP 2008. He has served as President of the European Association for Signal Processing (EURASIP) and he is currently a member of the Board of Governors for the IEEE CAS Society. He is the coauthor of four papers that have received best paper awards including the 2009 IEEE Computational Intelligence Society Transactions on Neural Networks Outstanding paper Award. He serves as an IEEE Signal Processing Society Distinguished Lecturer. He is a member of the Greek National Council for Research and Technology and Chairman of the SP advisory committee for the Edinburgh Research Partnership (ERP). He has served as vice chairman of the Greek Pedagogical Institute and for four years, he was a member of the Board of Directors of COSMOTE (the Greek mobile phone operating company). He is Fellow of IET and a Corresponding Fellow of FRSE.