Quaternion Reproducing Kernel Hilbert Spaces: Existence and Uniqueness Conditions

Felipe A. Tobar and Danilo P. Mandic, *Fellow, IEEE*

***Abstract*— The existence and uniqueness conditions of quaternion reproducing kernel Hilbert spaces (QRKHS) are established in order to provide a mathematical foundation for the development of quaternion-valued kernel learning algorithms. This is achieved through a rigorous account of left quaternion Hilbert spaces, which makes it possible to generalise standard RKHS to quaternion RKHS. Quaternion versions of the Riesz representation and Moore–Aronszajn theorems are next introduced, thus underpinning kernel estimation algorithms operating on quaternion-valued feature spaces. The difference between the proposed quaternion kernel concept and the existing real and vector approaches is also established in terms of both theoretical advantages and computational complexity. The enhanced estimation ability of the so-introduced quaternion-valued kernels over their real- and vector-valued counterparts is validated through kernel ridge regression applications. Simulations on real world 3D inertial body sensor data and nonlinear channel equalisation using novel quaternion cubic and Gaussian kernels support the approach.**

***Index Terms*— Quaternion RKHS, support vector regression, quaternion kernel ridge regression, high-dimensional kernels, vector kernels, multikernel.**

I. Introduction

SINCE their introduction in the early 1980s [1], support vector machines (SVM) have become a *de facto* standard for classification and regression with application in areas including semantic analysis [2], gene selection [3], and financial prediction [4]. An extension of the original SVM algorithm to nonlinear classification [5] employs the so-called *kernel trick* [6], which enables operation in higher dimensional spaces through the evaluation of a *kernel* function. The kernel trick has also made it possible to efficiently implement support vector regression (SVR) algorithms [7], whereby the output of a system is modelled as an inner product between a nonlinear transformation of the input and an optimal (fixed) weighting structure, in the so-called *feature space* [8], [9].

The theoretical justification for the use of the kernel trick is the reproducing property of the feature space. This is guaranteed by the Riesz representation theorem [10, Th. 6.2.4], which replaces the inner product of feature samples with kernel evaluations. Moreover, the Moore-Aronszajn theorem [11] ensures the existence and uniqueness of RKHS for positive definite (real or complex) kernels, thus replacing the cumbersome design of feature maps by a simple choice of kernels and combination parameters (weights). Dictated by the cost function, this class of algorithms includes kernel ridge regression (KRR) [12], kernel least mean square (KLMS) [13], kernel affine projection algorithm (KAPA) [14] and kernel recursive least squares (KRLS) [15].

Opportunities provided by the existing complex-valued RKHS framework [11] are manifold, however, for practical reasons related to the ease of tuning and the associated physical meaning, only real-valued Gaussian and polynomial kernels are typically considered. Kernel algorithms operating in complex-valued feature spaces are mostly of a complex kernel least mean square type [16]–[18]. Furthermore, higher-dimensional extensions of RKHS (in real vector-valued feature spaces [19], [20]) have also been developed and form the basis for both the multikernel least squares and multikernel least mean square [21], [22].

Recent developments in sensor technology have enabled routine recordings of 3D and 4D data (inertial body sensors, 3D wind modelling), this has been followed by the advances in linear estimation in the quaternion domain [23]–[28]. These have highlighted the usefulness of quaternion-valued algorithms to represent quadrivariate data. Quaternions [29] have already shown advantages over real-valued vectors within signal processing, computer graphics, and robotics communities, owing to their enhanced modelling of rotation, orientation, and cross-information between multichannel data. In addition, differential operators that enable gradient-based optimisation in the quaternion ring $\mathbb{H}$ have just been developed based on the $\mathbb{HR}$ calculus [30]. However, quaternion kernel estimation is still an emerging field [31], [32] and its development requires *rigorous existence and uniqueness conditions for quaternion reproducing kernel Hilbert spaces*, as these are pre-requisites to provide a theoretical basis for kernel algorithms operating in quaternion-valued feature spaces.

The aim of this work is to introduce the background theory and provide a proof-of-concept for quaternion-valued kernel estimation by establishing: (i) the existence and uniqueness properties of reproducing kernel Hilbert spaces of quaternion-valued functions, (ii) the notion of positive definiteness and reproducing property of quaternion RKHS, and (iii) the theoretical and practical differences between real, vector and quaternion RKHS. To this end, we first revisit the quaternion

Manuscript received April 22, 2013; revised February 4, 2014; accepted June 14, 2014. Date of publication June 30, 2014; date of current version August 14, 2014.

The authors are with the Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, U.K. (e-mail: f.tobar@imperial.ac.uk; d.mandic@imperial.ac.uk).

Communicated by G. Moustakides, Associate Editor for Detection and Estimation.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIT.2014.2333734

ring and quaternion left Hilbert spaces in order to define the quaternion RKHS (QRKHS). Quaternion versions of the Riesz representation [10] and Moore-Aronszajn [11] theorems are next presented (and compared to their real-valued counterparts) to introduce a unique relationship between QRKHSs and positive definite kernels. This equips us with a theoretical basis for quaternion kernel estimation, whereby the feature space has a corresponding quaternion-valued reproducing kernel.

We also show that the so-introduced QRKHS serves as a feature space of kernel regression algorithms within the kernel ridge regression setting. Applications in 3D body sensor tracking and nonlinear channel equalisation support the approach.

II. Quaternion Vector Spaces

A. The Quaternion Division Ring

The quaternion set $\mathbb{H}$ is a four-dimensional vector space over the real field $\mathbb{R}$ spanned by the linearly independent basis $\{1, i, j, k\}$ [29]. Accordingly, any element $q \in \mathbb{H}$ can be written as a linear combination $q = a1 + bi + cj + dk$, where $a, b, c, d \in \mathbb{R}$.

The sum and the scalar multiplication are defined in an element-wise fashion as in $\mathbb{R}^4$, that is

$$\begin{pmatrix} a_1 \\ b_1 \\ c_1 \\ d_1 \end{pmatrix} + \begin{pmatrix} a_2 \\ b_2 \\ c_2 \\ d_2 \end{pmatrix} = \begin{pmatrix} a_1 + a_2 \\ b_1 + b_2 \\ c_1 + c_2 \\ d_1 + d_2 \end{pmatrix}$$
$$\alpha(a, b, c, d) = (\alpha a, \alpha b, \alpha c, \alpha d), \quad \alpha \in \mathbb{R} \qquad (1)$$

where the notation $(a, b, c, d) = (a, b, c, d)^T = a1 + bi + cj + dk \in \mathbb{H}$ is used for convenience of presentation.

Remark 1: The pair $(\mathbb{H}, +)$ is an Abelian group [10], for which the addition operation is defined in (1) and the additive identity is $0 = (0, 0, 0, 0) \in \mathbb{H}$.

The *quaternion multiplication* (or *Hamilton product*) is a bilinear mapping $\mathbb{H} \times \mathbb{H} \to \mathbb{H}$, $(p, q) \mapsto pq$, defined by

$$pq = \begin{pmatrix} a_1 \\ b_1 \\ c_1 \\ d_1 \end{pmatrix} \begin{pmatrix} a_2 \\ b_2 \\ c_2 \\ d_2 \end{pmatrix} = \begin{pmatrix} a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2 \\ a_1b_2 + b_1a_2 + c_1d_2 - d_1c_2 \\ a_1c_2 - b_1d_2 + c_1a_2 + d_1b_2 \\ a_1d_2 + b_1c_2 - c_1b_2 + d_1a_2 \end{pmatrix}. \qquad (2)$$

Remark 2: The quaternion product defined in (2) distributes over the sum, i.e. $\forall p, q, r \in \mathbb{H}$

$$p(q + r) = pq + pr$$
$$(p + q)r = pr + qr.$$

It is also possible to express the quaternion multiplication using the basis expansion representation, that is, $(a_1 1 + b_1 i + c_1 j + d_1 k)(a_2 1 + b_2 i + c_2 j + d_2 k)$, and applying the multiplication rule

$$i^2 = j^2 = k^2 = ijk = -1.$$

Note that the basis element $1 = (1, 0, 0, 0) \in \mathbb{H}$ is the multiplicative identity, meaning that $q1 = 1q = q, \forall q \in \mathbb{H}$, and is therefore omitted in the basis representation, $q = a + bi + cj + dk$. We refer to the factor of $(1, 0, 0, 0)$ as *real part of q*, denoted by $\Re\{q\} = a$, and to the remaining factors as the *imaginary part of q*, denoted by $\Im\{q\} = (0, b, c, d)$.

For any given element $q \in \mathbb{H}$, $q \neq 0$, its multiplicative inverse $q^{-1} \in \mathbb{H} \setminus \{0\}$ is given by

$$q^{-1} = \frac{q^*}{\|q\|^2},$$

where $q^* = (a, -b, -c, -d)$ denotes the conjugate of q, and $\|q\| = \sqrt{q^*q} = \sqrt{qq^*} = \sqrt{a^2 + b^2 + c^2 + d^2}$ denotes the norm in $\mathbb{H}$ defined as the Euclidean norm in $\mathbb{R}^4$; as a consequence, $qq^{-1} = q^{-1}q = 1, \forall q \neq 0$. By using the conjugate operator, the real and imaginary parts of $q \in \mathbb{H}$ can be written respectively as

$$\Re\{q\} = \frac{q + q^*}{2}, \quad \Im\{q\} = \frac{q - q^*}{2}.$$

Remark 3: The pair $(\mathbb{H}, \cdot)$ equipped with the identity element is a monoid under multiplication, while the inclusion of the multiplicative inverse makes $(\mathbb{H}\setminus\{0\}, \cdot)$ a group [33], [34].

Remark 4: Since $(\mathbb{H}, +)$ is an Abelian group (Remark 1), $(\mathbb{H}, \cdot)$ is a group (Remark 3), and the quaternion product distributes over the sum (Remark 2), the triplet $(\mathbb{H}, +, \cdot)$ is a non-commutative division ring [34].

Despite the lack of commutativity in $\mathbb{H}$, its division ring properties establish the basis for the design of estimation algorithms. Furthermore, $\mathbb{H}$ is one of the four normed division algebras over the real field, the other three being the real field $\mathbb{R}$, the complex field $\mathbb{C}$, and the non-associative unitary octonion ring $\mathbb{O}$ (see the Frobenius theorem [35]).

B. Quaternion-Valued Hilbert Spaces

To introduce the concept of quaternion Hilbert space, we first need to define quaternion vector spaces and their algebraic properties.

Since $(\mathbb{H}, +, \cdot)$ is a division ring **and not a field** (it lacks the commutativity property), strictly speaking it is not possible to construct a general vector space over $\mathbb{H}$; however, we can still construct a *left-module*. A module [36], [37] is a generalisation of vector space which allows for the scalar set to be a ring (rather than a field). We refer to a left-module $\mathcal{H}$ over $\mathbb{H}$ as vector space [10] in which the **non-commutative** scalar multiplication $\mathbb{H} \times \mathcal{H} \to \mathcal{H}$ is defined on the left-hand side by $(q, \mathbf{x}) \mapsto q\mathbf{x}$.

We next set out to restate the concepts of inner product and left Hilbert space for quaternions, as these are required to define quaternion-valued RKHSs.

Definition 1 (Quaternion Left Hilbert Space): A nonempty set $\mathcal{H}$ is called a quaternion left Hilbert space if it is a quaternion left module (i.e. built over $\mathbb{H}$) and there exists a quaternion-valued function $\langle \cdot, \cdot \rangle : \mathcal{H} \times \mathcal{H} \to \mathbb{H}$ with the following properties:

1) *Conjugate symmetry:* $\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{y}, \mathbf{x} \rangle^*$.
2) *Linearity:* $\langle p\mathbf{x} + q\mathbf{y}, \mathbf{z} \rangle = p \langle \mathbf{x}, \mathbf{z} \rangle + q \langle \mathbf{y}, \mathbf{z} \rangle$.
3) *Conjugate linearity:* $\langle \mathbf{x}, p\mathbf{y} + q\mathbf{z} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle p^* + \langle \mathbf{x}, \mathbf{z} \rangle q^*$.
4) $\langle \mathbf{x}, \mathbf{x} \rangle \geq 0$ *and* $\langle \mathbf{x}, \mathbf{x} \rangle = 0 \iff \mathbf{x} = 0$.
5) *Completeness: If $\{\mathbf{x}_n\} \subset \mathcal{H}$ is a Cauchy sequence, then $\mathbf{x} = \lim_{n \to \infty} \mathbf{x}_n \in \mathcal{H}$.*

We refer to the function $\langle \cdot, \cdot \rangle$ as inner product and denote its induced norm by $\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}$.

Observation 1: The space $\mathbb{H}^n$, with the inner product $\langle p, q\rangle = p^T q^*$ is a quaternion left Hilbert space.

Observation 2: The space of quaternion-valued square-integrable functions $L_2 = \{f : X \in \mathbb{H}^n \to \mathbb{H}, s.t. \int_X \|f(\mathbf{x})\|^2 d\mathbf{x} < \infty\}$ with the inner product $\langle f, g\rangle = \int_X f(\mathbf{x}) g^*(\mathbf{x}) d\mathbf{x}$ is a quaternion left Hilbert space.[1]

Standard properties of real and complex Hilbert spaces such as the Cauchy-Schwarz inequality and the concept of orthogonality also extend to quaternion Hilbert spaces. In particular, we highlight two properties that will be helpful in the next section:

- The elements $\mathbf{x}, \mathbf{y} \in \mathcal{H}$ are orthogonal, denoted by $\mathbf{x} \perp \mathbf{y}$, if and only if $\langle \mathbf{x}, \mathbf{y}\rangle = 0$.
- Two sets $A, B \in \mathcal{H}$ are orthogonal if and only if $\mathbf{x} \perp \mathbf{y}$, $\forall \mathbf{x} \in A, \mathbf{y} \in B$. We denote by $A^\perp$ the set of all elements that are orthogonal to $\mathbf{x} \in A$.

For the properties of complex Hilbert spaces which also apply to the introduced left quaternion Hilbert space see [10].

III. Quaternion Reproducing Kernel Hilbert Spaces

We now introduce quaternion reproducing kernel Hilbert spaces to provide both theoretical support and physical insight for the design and implementation of quaternion-valued kernel estimation algorithms.

Definition 2 (Quaternion Reproducing Kernel Hilbert Space): Let X be an arbitrary set and $\mathcal{H}$ a left quaternion Hilbert space of functions from X to $\mathbb{H}$. We say that $\mathcal{H}$ is a quaternion reproducing kernel Hilbert space (QRKHS) if the (linear) evaluation map

$$\begin{aligned} L_{\mathbf{x}} : \mathcal{H} &\longrightarrow \mathbb{H} \\ f &\longmapsto f(\mathbf{x}) \end{aligned} \tag{3}$$

is bounded $\forall \mathbf{x} \in X$.

A. Riesz Representation Theorem

We can now introduce the following theorem in order to guarantee the existence of a reproducing kernel for any given QRKHS.

Theorem 1 (Quaternion Riesz Representation Theorem): For every bounded linear functional L defined over a quaternion left Hilbert space $\mathcal{H}$, there exists a unique element $g \in \mathcal{H}$ such that $L(f) = \langle f, g\rangle, \forall f \in \mathcal{H}$.

Proof: The proof follows from [10, Th. 6.2.4] and the properties of the inner product in quaternion left Hilbert spaces stated in Definition 1.

Denote by $A = \{f \in \mathcal{H} : L(f) = 0\}$ the null space of L. By continuity[2] of L, A is a closed linear subspace of $\mathcal{H}$. If $A = \mathcal{H}$, then $L = 0$ and $L(f) = \langle f, 0\rangle$. If $A \neq \mathcal{H}$, then there exists at least one element $g_0 \in \mathcal{H}$, such that $g_0 \neq 0$ and $g_0 \in A^\perp$ [10, Corollary 6.2.3]. By definition of g_0, $L(g_0) \neq 0$, and for any $f \in \mathcal{H}$ the element $f - L(f)\big(L(g_0)\big)^{-1} g_0 \in A$. As a consequence,

$$\left\langle f - L(f)\big(L(g_0)\big)^{-1} g_0, g_0 \right\rangle = 0.$$

Applying the properties of the inner product space we have

$$L(f)\big(L(g_0)\big)^{-1} \langle g_0, g_0\rangle = \langle f, g_0\rangle,$$

then, replacing $\langle g_0, g_0\rangle = \|g_0\|^2$ and right-multiplying both sides by $\frac{L(g_0)}{\|g_0\|^2}$ yields

$$L(f) = \langle f, g_0\rangle \frac{L(g_0)}{\|g_0\|^2}. \tag{4}$$

Now, by denoting $g = L^*(g_0) g_0 / \|g_0\|^2$ we arrive at the desired $L(f) = \langle f, g\rangle$.

To prove uniqueness, assume $g_1, g_2 \in \mathcal{H}$ such that $L(f) = \langle f, g_1\rangle = \langle f, g_2\rangle$. Therefore, $\langle f, g_1 - g_2\rangle = 0$ for all $f \in \mathcal{H}$; in particular, by taking $f = g_1 - g_2$ we have $\|g_1 - g_2\|^2 = 0 \Rightarrow g_1 = g_2$. □

Remark 5: Observe that the right-multiplication by $\frac{L(g_0)}{\|g_0\|^2}$, which yields Eq. (4), holds the key to differentiate the proof for Thm. 1 from that of the complex and real cases. Due to the non-commutative property of the quaternion ring, the element $g = L^*(g_0) g_0 / \|g_0\|^2$ is different from $\bar{g} = g_0 L^*(g_0) / \|g_0\|^2$, which is used in the proof for the real/complex cases in [10, Theorem 6.2.4].

Corollary 1 (Reproducing Property): For any $f \in \mathcal{H}$, there exists a unique element $K_{\mathbf{x}} \in \mathcal{H}$ such that the evaluation map $L_{\mathbf{x}} = f(\mathbf{x})$ in (3) can be expressed as $L_{\mathbf{x}} = \langle f, K_{\mathbf{x}}\rangle$.

Proof: As $L_{\mathbf{x}}$ is itself a bounded linear operator, based on the quaternion Riesz representation theorem there exists an element $g \in \mathcal{H}$ such that $L_{\mathbf{x}}(f) = \langle f, g\rangle$. The element $g = g(\mathbf{x})$ is unique for a given functional $L_{\mathbf{x}}$, or equivalently, for a given $\mathbf{x} \in X$. Therefore, we can define $K_{\mathbf{x}} \triangleq g$ and write $L_{\mathbf{x}} = \langle f, K_{\mathbf{x}}\rangle$. □

Since $K_{\mathbf{x}}(\cdot) \in \mathcal{H}$, it can be evaluated for any $\mathbf{y} \in X$. This allows us to define

$$\begin{aligned} K : X \times X &\longrightarrow \mathbb{H} \\ (\mathbf{x}, \mathbf{y}) &\longmapsto K(\mathbf{x}, \mathbf{y}) = K_{\mathbf{x}}(\mathbf{y}), \end{aligned}$$

whereby the function K is referred to as the *reproducing kernel* of the QRKHS $\mathcal{H}$. Its existence and uniqueness properties are a direct consequence of the quaternion Riesz representation theorem (Theorem 1). Similarly to the standard real- and complex-valued cases, the reproducing property of K can be expressed as

$$\forall f \in \mathcal{H} \text{ and } \mathbf{x} \in X, \;\; f(\mathbf{x}) = \langle f, K_{\mathbf{x}}\rangle.$$

The following relationships are readily obtained by applying the reproducing property on the functions $K_{\mathbf{x}} = K(\mathbf{x}, \cdot) \in \mathcal{H}$ and $K_{\mathbf{y}} = K(\mathbf{y}, \cdot) \in \mathcal{H}$:

- $K(\mathbf{x}, \mathbf{y}) = \langle K_{\mathbf{x}}, K_{\mathbf{y}}\rangle = \langle K_{\mathbf{y}}, K_{\mathbf{x}}\rangle^* = K^*(\mathbf{x}, \mathbf{y})$.
- $K(\mathbf{x}, \mathbf{x}) = \langle K_{\mathbf{x}}, K_{\mathbf{x}}\rangle = \|K_{\mathbf{x}}\|^2 \geq 0$.
- $K_{\mathbf{x}} = 0 \iff f(\mathbf{x}) = \langle f, K_{\mathbf{x}}\rangle = 0, \;\; \forall f \in \mathcal{H}$.

We have therefore shown, through the quaternion Riesz representation theorem, that for an arbitrary QRKHS there exists a unique reproducing kernel. This makes it possible to

[1] We considered the quaternion norm $\|q\| = \sqrt{q^* q}$ and the Lebesgue measure $d\mathbf{x}$ in $\mathbb{H}^n$ defined in analogy to the Lebesgue measure in $\mathbb{R}^{4n}$ (for instance for $n = 1$, $d\mathbf{x} = d\mathbf{x}_r d\mathbf{x}_i d\mathbf{x}_j d\mathbf{x}_k$).

[2] A bounded linear operator between normed spaces is always continuous, see [10, Th. 4.4.2].

compute inner products in a quaternion-valued feature space using the kernel trick.

Observe that although Theorem 1 gives theoretical support for quaternion kernel estimation, it is far from being useful in practice on its own, since the design of a QRKHS suited for a specific task can be rather difficult. To this end, we next complement the Riesz representation theorem with the Moore-Aronszajn theorem, in order to show that any quaternion kernel (within a certain class of kernels) generates a unique QRKHS.

B. Moore-Aronszajn Theorem

In the real-valued case, the existence of a unique QRKHS generated by a positive definite kernel is ensured via either (i) the Mercer theorem [38], where the feature Hilbert space is spanned by the eigenfunctions of the kernel K, or (ii) the Moore-Aronszajn theorem, in which the feature Hilbert space is spanned by the functions $K_{\mathbf{x}} = K(\mathbf{x}, \cdot)$.

We now state two equivalent definitions of positive definiteness in order to introduce a key result in quaternion reproducing kernel Hilbert spaces: the quaternion Moore-Aronszajn theorem.

Definition 3 (Positive Definiteness - Integral Form): A Hermitian kernel $K(\mathbf{x}, \mathbf{y}) = K^*(\mathbf{y}, \mathbf{x})$ is positive definite on the set X iff for any integrable function $\theta : X \to \mathbb{H}$, $\theta \neq 0$, it obeys

$$\int_X \int_X \theta^*(\mathbf{x}) K(\mathbf{x}, \mathbf{y}) \theta(\mathbf{y}) \mathrm{d}\mathbf{x} \mathrm{d}\mathbf{y} > 0.$$

Definition 4 (Positive Definiteness - Matrix Form): A Hermitian kernel $K(\mathbf{x}, \mathbf{y}) = K^*(\mathbf{y}, \mathbf{x})$ is positive definite on the set X iff the kernel matrix $\mathbf{K}_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$ is positive definite for any choice of the set $S_{\mathbf{x}} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subset X$, $m \in \mathbb{N}$.

Theorem 2 (Quaternion Moore-Aronszajn Theorem): For any positive definite quaternion-valued kernel K defined over a set X, there exists a unique (up to an isomorphism) left quaternion Hilbert space of functions $\mathcal{H}$ for which K is a reproducing kernel.

Proof: The proof first generalises the idea behind the real-valued Moore-Aronszajn theorem [11], to show that the span of $K_{\mathbf{x}}$ is a QRKHS, and then presents the uniqueness proof.

(i) The span of $K_{\mathbf{x}}$ is a QRKHS. Define the set

$$\mathcal{H}_0 = \left\{ f \in \mathcal{F} : f = \sum_{i=0}^{n} \alpha_i K(\mathbf{x}_i, \cdot), \mathbf{x}_i \in X, \alpha_i \in \mathbb{H}, n \in \mathbb{N} \right\}$$

and the inner product between $f = \sum_{i=0}^{n} \alpha_i K(\mathbf{x}_i, \cdot)$ and $g = \sum_{i=0}^{m} \beta_i K(\mathbf{y}_i, \cdot)$ as

$$\langle f, g \rangle = \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i K(\mathbf{x}_i, \mathbf{y}_j) \beta_j^*. \tag{5}$$

Note that the inner product $\langle \cdot, \cdot \rangle$ satisfies the properties in Definition 1 and the set $\mathcal{H}_0$ is a left inner product space. Its closure, denoted by $\mathcal{H} = \overline{\mathcal{H}_0}$, equips $\mathcal{H}_0$ with the limits of all its Cauchy sequences $\{f_n\} \subset \mathcal{H}_0$. As the elements added to form the closure are also bounded (Cauchy sequences are convergent), the elements of $\mathcal{H}$ can be written in the form $f = \sum_{i=0}^{\infty} \alpha_i K(\mathbf{x}_i, \cdot)$.

Observe that the evaluation functional (3) over the so-defined set $\mathcal{H}$ is bounded. Indeed, using the Cauchy-Schwartz inequality and the quaternion Riesz theorem (Thm. 1) we have

$$\begin{aligned} |f(\mathbf{x})|_{\mathbb{H}} &= |\langle f, K_{\mathbf{x}} \rangle|_{\mathbb{H}} \leq \|f\|_{\mathcal{H}} \|K_{\mathbf{x}}\|_{\mathcal{H}} \\ &= \|f\|_{\mathcal{H}} \sqrt{K(\mathbf{x}, \mathbf{x})} < \infty. \end{aligned}$$

(ii) Uniqueness. Consider two spaces $\mathcal{H}$ and $\mathcal{G}$ for which K is a reproducing kernel, and recall that the equation

$$\langle K_{\mathbf{x}}, K_{\mathbf{y}} \rangle_{\mathcal{H}} = K(\mathbf{x}, \mathbf{y}) = \langle K_{\mathbf{x}}, K_{\mathbf{y}} \rangle_{\mathcal{G}}$$

holds over the span of $\{K_{\mathbf{x}}, \mathbf{x} \in X\}$. As the closure of the span is unique and the inner product is linear, we have $\mathcal{H} = \mathcal{G}$.

We have therefore shown that given an arbitrary positive definite quaternion kernel K, there is a (unique) complete quaternion inner product space (i.e. a left Hilbert space), for which the evaluation functional is bounded, conditions for a QRKHS. □

Remark 6: Due to the non-commutativity of $\mathbb{H}$, the inner product constructed in the proof of Thm. 2, Eq. (5), differs from the real/complex case in that it requires a particular form in order to fulfil the requirements of Definition 1.

The so-constructed inner product supports the reproducing property of the QRKHS, that is,

$$f(\mathbf{x}) = \sum_{i=0}^{\infty} \alpha_i K(\mathbf{x}_i, \mathbf{x}) \overset{(a)}{=} \left\langle \sum_{i=0}^{\infty} \alpha_i K_{\mathbf{x}_i}, K_{\mathbf{x}} \right\rangle = \langle f(\cdot), K_{\mathbf{x}} \rangle,$$

where the symbol $\overset{(a)}{=}$ refers to the definition of the inner product in (5).

Theorems 1 and 2 provide the existence and uniqueness conditions underpinning quaternion-valued kernel algorithms: the Riesz representation theorem allows us to simplify feature space operations into kernel evaluations and the Moore-Aronzsajn theorem ensures that for any (positive definite) kernel there is a unique QRKHS.

Remark 7: Since the QRKHS is built upon a left-module, and not a field as the standard RKHS, the derivation of Theorems 1 and 2 confirms that commutativity is not a requirement for constructing feature spaces over division rings and also paves the way for the study of relationships between QRKHS built over left- and right-modules. We would also like to emphasise that the aim of the proofs provided is not claim a radical difference between Theorems 1 and 2, and their real versions, but to show that although the corresponding proofs follow the same criteria, the quaternion case requires more attention due to the lack of commutativity.

IV. Comparison to Other High-Dimensional Kernels

We shall now revisit existing approaches to high-dimensional kernels in order to highlight the differences between our proposed quaternion kernel framework and those based upon existing high-dimensional kernels. We address these differences in both theoretical and computational-complexity terms.

A. Vector-Valued RKHS (Matrix-Valued Kernels)

Unlike the original RKHS theory, where the elements of the feature space are (scalar) real-valued functions, the feature space can be constructed using the basis $\{\Psi(\mathbf{x}) : X \longrightarrow \mathbb{R}^n, \mathbf{x} \in X\}$, therefore yielding a vector-valued RKHS (VRKHS) with matrix-valued reproducing kernels $K(\mathbf{x}, \mathbf{y}) = \Psi^H(\mathbf{x})\Psi(\mathbf{y}) \in \mathbb{R}^{n\times n}$. The VRKHS theory was introduced in [39] and [40], and recent results exploiting this concept includes multi-category classification and multivariate regression [19], [20].

The matrix-valued kernel concept is particularly useful in multivariate regression, whereby the desired output is vector-valued. For illustration, consider the problem of learning the mapping $\mathbf{y} = f(\mathbf{x}) \in \mathbb{R}^n$, using[3] $\hat{\mathbf{y}} = \Psi^H(\mathbf{x})M$ where $\Psi(\cdot)$ forms the basis of the VRKHS $\mathcal{H}_V$. Following the classic kernel regression paradigm, M can be approximated according to the signal subspace principle [41], and the available measurements $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1:N}$, by $\widehat{M} = \sum_{i=1}^{N} \Psi(\mathbf{x}_i)\mathbf{A}_i$, where the (vector) weights $\{\mathbf{A}_i\}_{i=1:N} \subset \mathbb{R}^n$ are set according to the chosen optimisation criteria. Consequently, the estimate can be expressed as

$$\hat{\mathbf{y}} = \sum_{i=1}^{N} \Psi^H(\mathbf{x})\Psi(\mathbf{x}_i)\mathbf{A}_i = \sum_{i=1}^{N} K(\mathbf{x}, \mathbf{x}_i)\mathbf{A}_i$$

where $(\cdot)^H$ denotes the conjugate transpose (Hermitian) operator and $K(\mathbf{x}, \mathbf{x}_i) \in \mathbb{R}^{n\times n}$ and $\mathbf{A}_i \in \mathbb{R}^{n\times 1}$.

The vector-valued RKHS is physically meaningful in that it assumes that the dimensionality of the feature space is given by the dimensionality of the output of the regression problem. However, the matrix-valued nature of the resulting kernels hinders the implementation of multichannel regression, because both the number of kernel evaluations and multiplications grow quadratically with the size of the output vector. Indeed, as the kernel $K(\mathbf{x}_i, \mathbf{x})$ is symmetric, it contains $\frac{(n^2+n)}{2}$ different entries and the product $K(\mathbf{x}_i, \mathbf{x})\mathbf{A}_i$ involves n^2 multiplications. The fact that, within matrix-valued kernel regression, the dimension of the feature space is dictated by the dimension of the output contradicts the fundamental concept of using feature spaces (of a different dimension) to explain the observed relationship in the data. See Table I for the computational complexity of other higher-dimensional kernel algorithms.

We next review another real-valued higher dimensional approach for kernel regression, referred to as *multikernel* or *vector kernel*, which allows for the use of multiple kernels. As desired, the dimensionality of the vector kernel is set as a design parameter, and does not depend on the the number of channels of the input/output data.

B. Multikernel Learning

The multikernel concept also uses a VRKHS as feature space and redefines the inner product [21] so as to only preserve the diagonal elements of the matrix-valued product $\Psi^H\Psi$. This yields a vector-valued kernel $\overrightarrow{K}(\mathbf{x}, \mathbf{y}) = \text{diag}\left(\Psi^H(\mathbf{x})\Psi(\mathbf{y})\right) \in \mathbb{R}^L$, where L is the number of subkernels. The regression estimate corresponding to this vector-valued kernel can be expressed by

$$\hat{\mathbf{y}} = \sum_{i=1}^{N} A_i^T \overrightarrow{K}(\mathbf{x}, \mathbf{x}_i),$$

where $A_i \in \mathbb{R}^{L\times n}$, $\hat{\mathbf{y}} \in \mathbb{R}^n$. Additionally, by denoting the entries of the vector-valued kernel (subkernels) as $\overrightarrow{K} = [K_1, K_2, \ldots, K_L]^T$, and the coefficients $A_i = [a_{i,1}, a_{i,2}, \ldots, a_{i,L}]^T$, it is possible to express the vector product as a summation, giving the estimate in the explicit multiple-kernel (multikernel) format

$$\hat{\mathbf{y}} = \sum_{j=1}^{L}\sum_{i=1}^{N} a_{ij} K_j(\mathbf{x}, \mathbf{x}_i),$$

where $a_{ij} \in \mathbb{R}^n$ and $K_j(\mathbf{x}, \mathbf{x}_i) \in \mathbb{R}$.

The structure of the multikernel concept is intuitive and its ability to capture *different types* of nonlinear behaviour from the input data has been documented in [21], [22], [42], and [43]. Furthermore, the approach is flexible, since the number of subkernels does not depend on the dimension of the input or output data (L is not necessarily equal to n), but is only set as a design parameter based on the observed nonlinear features of the data and the available computational power.

The following lemma gives a sufficient condition designing for vector-kernels.

Lemma 1: $\overrightarrow{K} = [K_1, K_2, \ldots, K_L]^T$ is a valid vector-valued kernel if all its subkernels are positive definite scalar kernels.

Proof: The proof follows from the construction of the vector-kernel. If every subkernel K_i is positive definite, then there exists a mapping ψ_i such that $K_i(\mathbf{x}, \mathbf{y}) = \psi_i^H(\mathbf{x})\psi_i(\mathbf{y})$. As a consequence, by denoting $\Psi = [\psi_1, \ldots, \psi_L]$ the array of the mappings ψ_i, we have $\overrightarrow{K}(\mathbf{x}, \mathbf{y}) = \text{diag}\left(\Psi^H(\mathbf{x})\Psi(\mathbf{y})\right)$, that is, a vector-valued kernel. □

Lemma 1 represents the backbone of the multikernel concept: in multiple kernel learning [42], the output is approximated by a sum of partial approximations using subkernels, hence, the estimate can be seen as an ensemble of estimators in which each stage (subkernel) is responsible of estimating *one type of nonlinearity*. An example of multikernel regression which employs an LMS update strategy to predict wind can be found in [21], where each subkernel within the multikernel approach accounts for different dynamical properties of the 3D wind. For additional insight into multikernel learning see [22], [42], [44].

C. Quaternion Kernels

The introduced quaternion-valued feature spaces allow us to adopt the standard regression formulation, that is,

$$\hat{\mathbf{y}} = \sum_{i=1}^{N} a_i K(\mathbf{x}, \mathbf{x}_i), \;\; a_i \in \mathbb{H}.$$

[3]For $\mathbf{x} \in X$, $\Psi(\mathbf{x}) \in \mathbb{R}^{|X|\times n}$, and the coefficient $M \in \mathbb{R}^{|X|\times 1}$, where $|X|$ is the (possibly infinite) cardinality of X. Observe that $\mathbb{R}^{|X|\times 1}$ is the space of linear mappings between the empirical feature space $\left\{\Psi(\mathbf{x}) \in \mathbb{R}^{|X|\times n}, \mathbf{x} \in X\right\}$ and $\mathbb{R}^n$.

The following lemma states properties of quaternion positive definite kernels.

Lemma 2: Let $K = K_r + iK_i + jK_j + kK_k$ be a quaternion kernel, then

(a) *K is Hermitian iff K_r is symmetric positive definite and $K_i = -K_i^T$, $K_j = -K_j^T$ and $K_k = -K_k^T$.*

(b) *If K is Hermitian, K is positive definite iff the **real-valued** matrix representation[4] of its Gram matrix is positive definite.*

Proof: **(a)** The Hermitian condition of the quaternion kernel $K = K^H$ can be expressed in terms of its real an imaginary parts as

$$K_r + iK_i + jK_j + kK_k = K_r^T - iK_i^T - jK_j^T - kK_k^T.$$

Therefore, since $\{1, i, j, k\}$ is a linearly-independent basis of $\mathbb{H}$, the above relationship is true if and only if the corresponding terms are equal coordinate-wise, that is:

$$K_r = K_r^T, K_i = -K_i^T, K_j = -K_j^T, K_k = -K_k^T.$$

(b) Let us now re-state the matrix definition (Definition 3) of positive definiteness $\mathbf{v}^H\mathbf{K}\mathbf{v} > 0, \forall \mathbf{v} \in \mathbb{H}^m \setminus \{0\}$ as

$$\Re\{\mathbf{v}^H\mathbf{K}\mathbf{v}\} > 0 \tag{6}$$

$$\Im\{\mathbf{v}^H\mathbf{K}\mathbf{v}\} = 0. \tag{7}$$

where $\mathbf{K} \in \mathbb{H}^{m\times m}$ is the Gram matrix corresponding to an arbitrary collection of m points $\{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subset X$. Observe that Eq. (6) is a necessary and sufficient condition for a Hermitian quaternion kernel $\mathbf{K}$ to be positive definite, since (7) always holds due to the Hermitian property of the kernel:

$$2\Im\{\mathbf{v}^H\mathbf{K}\mathbf{v}\} = \mathbf{v}^H\mathbf{K}\mathbf{v} - (\mathbf{v}^H\mathbf{K}\mathbf{v})^H = \mathbf{v}^H\mathbf{K}\mathbf{v} - (\mathbf{v}^H\mathbf{K}\mathbf{v}) = 0.$$

To analyse (6) in terms of real and imaginary parts of the quaternion kernel, we expand the vector $\mathbf{v} = \mathbf{v}_r + i\mathbf{v}_i + j\mathbf{v}_j + k\mathbf{v}_k$ and the kernel matrix $\mathbf{K} = \mathbf{K}_r + i\mathbf{K}_i + j\mathbf{K}_j + k\mathbf{K}_k$ using their real and imaginary parts and rewrite Eq. (6) as

$$\begin{aligned}\Re\{\mathbf{v}^H\mathbf{K}\mathbf{v}\} = {} & \mathbf{v}_r^T\mathbf{K}_r\mathbf{v}_r + \mathbf{v}_i^T\mathbf{K}_r\mathbf{v}_i + \mathbf{v}_j^T\mathbf{K}_r\mathbf{v}_j + \mathbf{v}_k^T\mathbf{K}_r\mathbf{v}_k \\ & + 2\mathbf{v}_i^T\mathbf{K}_i\mathbf{v}_r + 2\mathbf{v}_j^T\mathbf{K}_j\mathbf{v}_r + 2\mathbf{v}_k^T\mathbf{K}_k\mathbf{v}_r \\ & + 2\mathbf{v}_i^T\mathbf{K}_j\mathbf{v}_k + 2\mathbf{v}_j^T\mathbf{K}_k\mathbf{v}_i + 2\mathbf{v}_k^T\mathbf{K}_i\mathbf{v}_j > 0. \end{aligned} \tag{8}$$

Observe that Eq. (8) can be written as a quadratic positive-definite form $\mathbf{r}_\mathbf{v}^T\mathbf{Q}\mathbf{r}_\mathbf{v} \geq 0$, where

$$\mathbf{r}_\mathbf{v} = \begin{pmatrix}\mathbf{v}_r \\ \mathbf{v}_i \\ \mathbf{v}_j \\ \mathbf{v}_k\end{pmatrix}, \quad \mathbf{Q} = \begin{pmatrix}\mathbf{K}_r & -\mathbf{K}_i & -\mathbf{K}_j & -\mathbf{K}_k \\ \mathbf{K}_i & \mathbf{K}_r & -\mathbf{K}_k & \mathbf{K}_j \\ \mathbf{K}_j & \mathbf{K}_k & \mathbf{K}_r & -\mathbf{K}_i \\ \mathbf{K}_k & -\mathbf{K}_j & \mathbf{K}_i & \mathbf{K}_r\end{pmatrix}$$

are respectively an $\mathbb{R}^{4m}$ representation of $\mathbf{v}$ and the real-valued matrix representation of the Gram matrix of K.

We have therefore proved that the positive definiteness condition of a quaternion kernel can be verified through the positive definiteness of its real-valued matrix representation and vice versa. ∎

[4]See [45, p. 91] for the real matrix representation of quaternions.

D. Connections Between High-Dimensional Kernels

We next show that the matrix-valued, vector-valued, and quaternion-valued kernels are not alternative representations of the same mapping; they are different classes of kernels and generate different feature spaces. For the matrix-valued kernel this follows from its dimension, which is given by the number of data channels of the output. Consequently, a matrix kernel cannot be designed to have an arbitrary dimension as its quaternion- and vector-valued counterparts.

We now discuss whether a quaternion kernel and the particular case of a 4D vector kernel are two representations of the same mapping. Although these share the same number of degrees of freedom (four) and computational complexity, we show that these are different mappings through the following theorem.

Theorem 3: Let K and $\overrightarrow{K}$ be arbitrary quaternion- and vector-kernels given by

$$\begin{aligned} K &= K_r + iK_i + jK_j + kK_k \in \mathbb{H} \\ \overrightarrow{K} &= [K_1\ K_2\ K_3\ K_4]^T \in \mathbb{R}^4, \end{aligned}$$

where the real and imaginary parts of K, and the subkernels of $\overrightarrow{K}$ are scalar, real-valued, functions.

The following statements about the $\mathbb{R}^4$ representation of K and the quaternion representation of $\overrightarrow{K}$ are true:

(a) *$K' = [K_r\ K_i\ K_j\ K_k]^T \in \mathbb{R}^4$ is not a vector-kernel,*

(b) *$\overrightarrow{K}' = K_1 + iK_2 + jK_3 + kK_4 \in \mathbb{H}$ is not a Hermitian positive definite quaternion-kernel.*

Proof: The proof follows from the properties of vector and quaternion kernels stated in Lemmas 1 and 2.

(a) Quaternion kernels are Hermitian and positive definite; consequently, according to Lemma 2(a) the imaginary parts of K given by K_i, K_j, K_k are not symmetric. Therefore, based on Lemma 1, the array $K' = [K_r\ K_i\ K_j\ K_k]$ is not a vector-kernel (since its subkernels are not symmetric and positive definite).

(b) As $\overrightarrow{K}$ is a vector-kernel, Lemma 1 states that K_1, K_2, K_3 are symmetric and positive definite. Therefore, $\overrightarrow{K}' = K_1 + iK_2 + jK_3 + kK_4$ is not Hermitian due to Lemma 2(b). □

We now state the main consequence of Theorem 3. Although the $\mathbb{R}^4$ representation of the QRKHS generated by a quaternion kernel K of the form

$$\mathcal{H}_\mathbb{R} = \left\{ \begin{pmatrix}\Re f(\mathbf{x}) \\ \Im_i f(\mathbf{x}) \\ \Im_j f(\mathbf{x}) \\ \Im_k f(\mathbf{x})\end{pmatrix} \in \mathbb{R}^4, \text{ s.t. } f(\mathbf{x}) = \sum_{n=1}^{\infty} a_n K(\mathbf{x}_n, \mathbf{x}) \right\} \tag{9}$$

where $a_n \in \mathbb{H}$, $\mathbf{x} \in \mathbf{X}$, is indeed an RKHS (its evaluation functional is bounded), its real vector-valued reproducing kernel is not given by the $\mathbb{R}^4$ representation of the quaternion kernel $K' = [K_r\ K_i\ K_j\ K_k]$ of $K \in \mathbb{H}$, because K' is not a vector kernel in $\mathbb{R}^4$.

Remark 8: The vector-valued representation of a QRKHS in Eq. (9) is a VRKHS, however, finding its associated vector-valued kernel is not trivial.

TABLE I

NUMBER OF OPERATIONS REQUIRED TO EVALUATE THE TERM $AK(\mathbf{x}_s, \mathbf{x})$ FOR AN n-DIMENSIONAL OUTPUT AND A SINGLE SUPPORT VECTOR $\mathbf{x}_s$. THE COMPARISON IS PRESENTED FOR REAL-VALUED (SCALAR), QUATERNION-VALUED (SCALAR), VECTOR-VALUED ($\mathbb{R}^4$-VALUED), AND MATRIX-VALUED ($\mathbb{R}^{n\times n}$ – ACCORDING TO THE OUTPUT) KERNELS. THE TERMS κ_R, κ_Q DENOTE RESPECTIVELY REAL AND QUATERNION KERNEL EVALUATIONS WHILE μ_R, μ_Q DENOTE SCALAR MULTIPLICATIONS. THE COMPUTATIONAL COMPLEXITY DUE TO SUMMATIONS IS OMITTED AS IT IS NEGLIGIBLE COMPARED TO MULTIPLICATIONS AND KERNEL EVALUATIONS

Kernel	Scalar	Quaternion	Multikernel	Matrix-kernel
Kernel evaluations	$1\kappa_R$	$1\kappa_Q$	$4\kappa_R$	$\frac{n^2+n}{2}\kappa_R$
Multiplications	$n\mu_R$	$n\mu_Q$	$4n\mu_R$	$n^2\mu_R$
Total operations	$\kappa_R + n\mu_R$	$\kappa_Q + n\mu_Q$	$4(\kappa_R + n\mu_R)$	$\frac{n^2+n}{2}\kappa_R + n^2\mu_R$

E. Computational Complexity

Table I illustrates the computational complexity of kernel regression algorithms for scalar, quaternion, multikernel and matrix-kernel approaches. Observe that scalar kernels are simplest, whereas the complexity of matrix kernels grows quadratically with the dimension of the signal.

In order to compare the complexity associated to vector- and quaternion-valued kernels, observe that (i) μ_R involves four pure real multiplications (real kernels, quaternion weights) while μ_Q involves 16 (quaternion valued kernels and weights), and (ii) the quaternion kernel evaluation can be seen as the evaluation of four real functions. Consequently, by assuming $\kappa_Q \sim 4\kappa_R$, $\mu_Q \sim 4\mu_R$, Table I shows that the computational complexities of both kernel and quaternion kernel algorithms are of the same order.

Therefore, the implementation of both vector and quaternion kernels can benefit from exploiting the similarity among sub-kernels, as well as from an efficient framework to perform quaternion operations [46].

V. DESIGN OF QUATERNION-VALUED MERCER KERNELS

Theorem 2 gives the justification for the design and implementation of nonlinear kernel algorithms operating in QRKHS to simplify into the choice of a positive-semidefinite kernel. We next introduce and analyse the properties of some specific kernels of quaternion variable and justify their use within quaternion SVR algorithms.

A. Polynomial Kernel: The Quaternion Cubic Example

The polynomial kernel is standard in kernel-based estimation due to its robustness and ease of implementation. For real- and complex-valued samples $\mathbf{x}_r, \mathbf{y}_r$, the polynomial kernel is given by

$$K_P(\mathbf{x}_r, \mathbf{y}_r) = \left(1 + \mathbf{x}_r^T \mathbf{y}_r\right)^p$$

where $p \in \mathbb{N}$ is referred to as the order of the kernel. On the other hand, the real-valued polynomial kernel of quaternion samples $\mathbf{x}, \mathbf{y}$ $K_{RP} : X^2 \to \mathbb{R}$, that is, the polynomial kernel of the real-valued representations of $\mathbf{x}$ and $\mathbf{y}$, can be expressed as

$$K_{RP}(\mathbf{x}, \mathbf{y}) = \left(1 + \langle \mathbf{x}, \mathbf{y} \rangle_{\Re}\right)^p = \left(1 + \Re\{\mathbf{x}^H \mathbf{y}\}\right)^p \qquad (10)$$

where $\langle \mathbf{x}, \mathbf{y} \rangle_{\Re}$ is the inner product in $\mathbb{R}^n$ and $\Re\{q\}$ denotes the real part of the quaternion q.

The extension to quaternion-valued polynomial kernels is not straightforward, as for the quaternion vectors $\mathbf{x}$ and $\mathbf{y}$ the factorisation

$$\left(1 + \mathbf{x}^H \mathbf{y}\right)^p = \phi^H(\mathbf{x})\phi(\mathbf{y})$$

may not be possible due to the noncommutativity of the quaternion ring, and therefore the positive definiteness of such kernel cannot be guaranteed in this manner.

For $p = 3$, we next propose a quaternion polynomial kernel which admits factorisation as an inner product, thus ensuring the required positive definiteness.

Consider the quaternion cubic kernel $K_{QP} : X^2 \to \mathbb{H}$ given by

$$K_{QP}(\mathbf{x}, \mathbf{y}) = \left(1 + \mathbf{x}^H \mathbf{x}\right)\left(1 + \mathbf{x}^H \mathbf{y}\right)\left(1 + \mathbf{y}^H \mathbf{y}\right). \qquad (11)$$

To show that K_{QP} is positive semidefinite, we shall first consider its factorisation of the form $K_{QP}(\mathbf{x}, \mathbf{y}) = \phi^H(\mathbf{x})\phi(\mathbf{y})$. Indeed,

$$\begin{aligned} K_{QP}(\mathbf{x}, \mathbf{y}) &= \left(1 + \mathbf{x}^H \mathbf{x}\right)\left(1 + \mathbf{y}^H \mathbf{y}\right) \\ &\quad + \left(1 + \mathbf{x}^H \mathbf{x}\right)\mathbf{x}^H \mathbf{y}\left(1 + \mathbf{y}^H \mathbf{y}\right) \\ &= \left(1 + \mathbf{x}^H \mathbf{x}\right)^H \left(1 + \mathbf{y}^H \mathbf{y}\right) \\ &\quad + \left(\mathbf{x}\left(1 + \mathbf{x}^H \mathbf{x}\right)\right)^H \left(\mathbf{y}\left(1 + \mathbf{y}^H \mathbf{y}\right)\right) \\ &= \phi_1^H(\mathbf{x})\phi_1(\mathbf{y}) + \phi_2^H(\mathbf{x})\phi_2(\mathbf{y}), \end{aligned}$$

where $\phi_1(\mathbf{x}) = 1 + \mathbf{x}^H \mathbf{x}$ and $\phi_2(\mathbf{x}) = \mathbf{x}\left(1 + \mathbf{x}^H \mathbf{x}\right)$. Therefore, by setting $\phi(\mathbf{x}) = [\phi_1^T(\mathbf{x})\ \phi_2^T(\mathbf{x})]^T$ we arrive at

$$K_{QP}(\mathbf{x}, \mathbf{y}) = \phi^H(\mathbf{x})\phi(\mathbf{y}). \qquad (12)$$

Finally, by combining (12) and Definition 3 we have

$$\begin{aligned} &\int_{X^2} \theta^*(\mathbf{x}) K_{QP}(\mathbf{x}, \mathbf{y}) \theta(\mathbf{y}) \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{y} \\ &\quad = \int_{X^2} \theta^*(\mathbf{x}) \phi^H(\mathbf{x}) \phi(\mathbf{y}) \theta(\mathbf{y}) \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{y} \\ &\quad \overset{(a)}{=} \int_X \theta^*(\mathbf{x}) \phi^H(\mathbf{x}) \mathrm{d}\mathbf{x} \int_X \phi(\mathbf{y}) \theta(\mathbf{y}) \mathrm{d}\mathbf{y} \\ &\quad = \left(\int_X \phi(\mathbf{x}) \theta(\mathbf{x}) \mathrm{d}\mathbf{x}\right)^H \int_X \phi(\mathbf{y}) \theta(\mathbf{y}) \mathrm{d}\mathbf{y} \\ &\quad = \left\| \int_X \phi(\mathbf{x}) \theta(\mathbf{x}) \mathrm{d}\mathbf{x} \right\|^2 \geq 0, \end{aligned}$$

where $\theta(\cdot)$ is assumed to be Lebesgue integrable and bounded on the compact set $X \subsetneq \mathbb{H}^n$, while the identity $\overset{(a)}{=}$ is a consequence of Fubini's theorem [10].

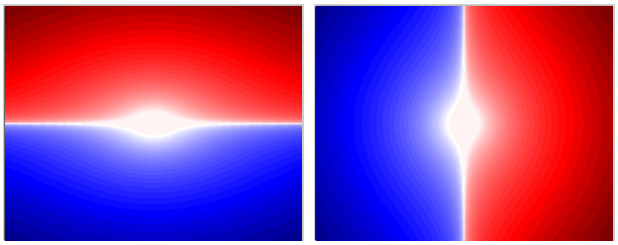

Fig. 1. Real (left) and i-imaginary (right) parts of K_{QP}. The colourmap is dark blue for $-13 \cdot 10^3$, white for the interval $[-10, 10]$, and red for $13 \cdot 10^3$ with a logarithmic RGB interpolation.

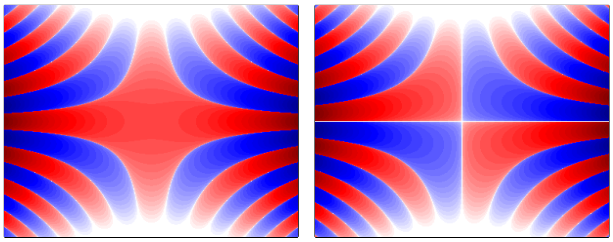

Fig. 2. Real (left) and i-imaginary (right) parts of K_{QG}. The colourmap is dark blue for $-7 \cdot 10^{-4}$, white for 0, and red for $7 \cdot 10^4$, with a logarithmic RGB interpolation.

Remark 9: Note from Eqs. (10) and (11) that, owing to its imaginary part, the quaternion cubic kernel K_{QP} provides enhanced data representation over the real-valued cubic kernel K_{RP}. Therefore, K_{QP} has the ability to learn the relationship between input variables, while preserving the mathematical simplicity of polynomial kernels.

Fig. 1 visualises K_{QP} for the scalar case $x = 1, y = y_r + iy_i + jy_j + ky_k \in \mathbb{H}$, which gives $K_{QP}(1, y) = 2(1 + y)(1 + \|y\|^2)$. As $K_{QP}(1, y)$ is symmetric, we only plot the region $(y_r, y_i, y_j, y_k) \in [-15, 15] \times [-15, 15] \times \{0\} \times \{0\}$.

B. Real-Valued Gaussian Kernel

Together with the polynomial kernel, the Gaussian kernel is extensively used in machine learning and signal processing applications. It serves as a deviation measure between samples, hence, providing reliable estimates for known regions of the input space (dictionary), while vanishing for inputs that deviate from the dictionary. The Gaussian kernel can be extended to operate on quaternion samples by accommodating the quaternion norm in its argument. Recall that $\|q\| = \sqrt{q^H q}$, so that the real-valued Gaussian kernel K_{RG} can be defined as

$$K_{RG}(\mathbf{x}, \mathbf{y}) = \exp\left(-A_R\,(\mathbf{x} - \mathbf{y})^H\,(\mathbf{x} - \mathbf{y})\right), \qquad (13)$$

where $A_R > 0$ is the kernel parameter.

We next use Definition 4 to show that K_{RG} is positive definite in the quaternion domain. First, observe that for an arbitrary, non-zero, vector $\mathbf{x} \in \mathbb{H}^n$ the quadratic form $\mathbf{x}^H\mathbf{K}\mathbf{x}$ is real. Indeed, due to the symmetry of the real matrix $\mathbf{K}$ we have

$$2\Im\{\mathbf{x}^H\mathbf{K}\mathbf{x}\} = \mathbf{x}^H\mathbf{K}\mathbf{x} - (\mathbf{x}^H\mathbf{K}\mathbf{x})^H = \mathbf{x}^H\mathbf{K}\mathbf{x} - (\mathbf{x}^H\mathbf{K}\mathbf{x}) = 0.$$

Now, by expanding the vector $\mathbf{x} = \mathbf{x}_r + i\mathbf{x}_i + j\mathbf{x}_j + k\mathbf{x}_k$ within $\Re\{\mathbf{x}^H\mathbf{K}\mathbf{x}\}$ using its real and imaginary parts, we can write

$$\Re\{\mathbf{x}^H\mathbf{K}\mathbf{x}\} = \mathbf{x}_r^T\mathbf{K}\mathbf{x}_r + \mathbf{x}_i^T\mathbf{K}\mathbf{x}_i + \mathbf{x}_j^T\mathbf{K}\mathbf{x}_j + \mathbf{x}_k^T\mathbf{K}\mathbf{x}_k.$$

Since $\mathbf{K}$ is positive definite in the real domain, the arbitrary components $\mathbf{x}_R, \mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k$ are real-valued, and the quadratic form $\mathbf{x}^H\mathbf{K}\mathbf{x}$ is positive, we have $\mathbf{x}^H\mathbf{K}\mathbf{x} = \Re\{\mathbf{x}^H\mathbf{K}\mathbf{x}\} > 0$, proving the positive definiteness of the real Gaussian kernel K_{RG} in $\mathbb{H}$.

C. Quaternion-Valued Gaussian Kernel

Recent complex-valued extensions of kernel estimation algorithms [17], [18] consider a complex-valued positive definite version of the real Gaussian kernel [47]. A quaternion version of such a kernel is

$$K_{QG}(\mathbf{x}, \mathbf{y}) = \exp\left(-A_Q\,(\mathbf{x} - \mathbf{y}^*)^T\,(\mathbf{x} - \mathbf{y}^*)\right), \qquad (14)$$

where $A_Q > 0$ is the kernel parameter.

Observe that the differences between K_{QG} and K_{RG} are in that K_{QG} is a function of $\mathbf{x} - \mathbf{y}^*$ (rather than $\mathbf{x} - \mathbf{y}$, hence allowing for the positive definiteness of the kernel), while the transpose (instead of the Hermitian) operator allows its argument to be a full quaternion.

By denoting $\tilde{\mathbf{e}}_R = \Re\{\mathbf{x} - \mathbf{y}^*\}$ and $\tilde{\mathbf{e}}_I = \Im\{\mathbf{x} - \mathbf{y}^*\}$, we write $\mathbf{x} - \mathbf{y}^* = \tilde{\mathbf{e}}_R + \tilde{\mathbf{e}}_I$ and expand K_{QG} according to

$$\begin{aligned} K_{QG}(\mathbf{x}, \mathbf{y}) &= \exp\left(-A_Q\,(\tilde{\mathbf{e}}_R + \tilde{\mathbf{e}}_I)^T\,(\tilde{\mathbf{e}}_R + \tilde{\mathbf{e}}_I)\right) \\ &= \exp\left(-A_Q\left(\tilde{\mathbf{e}}_R^T\tilde{\mathbf{e}}_R + \tilde{\mathbf{e}}_I^T\tilde{\mathbf{e}}_R + \tilde{\mathbf{e}}_R^T\tilde{\mathbf{e}}_I + \tilde{\mathbf{e}}_I^T\tilde{\mathbf{e}}_I\right)\right) \\ &= \exp\left(-A_Q\left(\|\tilde{\mathbf{e}}_R\|^2 - \|\tilde{\mathbf{e}}_I\|^2 + 2\tilde{\mathbf{e}}_R^T\tilde{\mathbf{e}}_I\right)\right) \\ &= e^{\delta}\left(\cos\|\Delta\| + \frac{\Delta}{\|\Delta\|}\sin\|\Delta\|\right), \end{aligned}$$

where $\delta = -A_Q\left(\|\tilde{\mathbf{e}}_R\|^2 - \|\tilde{\mathbf{e}}_I\|^2\right)$ and $\Delta = -2A_Q\tilde{\mathbf{e}}_R^T\tilde{\mathbf{e}}_I$.

Unlike the real Gaussian kernel K_{RG}, K_{QG} is not globally bounded since its norm grows exponentially with $\|\tilde{\mathbf{e}}_I\|^2 = \|\Im\{\mathbf{x}\} + \Im\{\mathbf{y}\}\|^2$ (as $A_Q > 0$). This highlights both advantages and disadvantages regarding the implementation of kernel estimation algorithms: K_{QG} has the ability to model data with large dynamics and to boost the speed of learning due to its exponential growth; however, an incorrect choice of parameters will lead to unbounded estimates. From the point of view of a physically-meaningful representation, the real Gaussian kernel K_{RG} is better suited for interpolation applications as it can be regarded as a measure of similarity of samples (like the triangular kernel in similarity-based modelling [48]), whereas the quaternion Gaussian kernel K_{QG} is useful for extrapolating nonlinear features.

Fig. 2 shows K_{QG} for the scalar case $\mathbf{y} = 0, \mathbf{x} = x_r + ix_i + jx_j + kx_k \in \mathbb{H}$, which gives $\delta = -A_Q(x_r^2 - x_i^2 - x_j^2 - x_k^2)$, $\Delta = -2A_Q x_r(ix_i + jx_j + kx_k)$. As $K_{QG}(x, 0)$ is symmetric, we only plot the region $(x_r, x_i, x_j, x_k) \in [-15, 15] \times [-15, 15] \times \{0\} \times \{0\}$ where $A_Q = 0.05$.

VI. Kernel Ridge Regression in Quaternion RKHS

We validated the proposed QRKHS, together with the introduced quaternion kernels, in the kernel ridge regression (KRR) setting presented in Section VI-A against both scalar and vector-valued real kernels; these results complement previous quaternion kernel applications which considered linear (quaternion) kernels only [31]. Section VI-B illustrates the prediction of body sensor signals and shows that the introduced quaternion cubic kernel outperforms the real vector-kernel (ensemble of four real cubic kernels), while having the same degrees of freedom and similar computational complexity. Section VI-C considers the nonlinear channel equalisation problem using real Gaussian and quaternion Gaussian kernels, and both linear and widely-linear estimators, in order to validate the quaternion Gaussian kernel generalisation (extrapolation) properties and robustness to overfitting. This second experiment also explains how to choose the parameters of the quaternion Gaussian kernel using available measurements.

A. Introduction to Kernel Ridge Regression

Consider the collection of available measurement pairs $C_N = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1:N} \in X \times Y$ corresponding to the input and output of a nonlinear system. The aim of KRR is to model the relationship between the input $\mathbf{x}$ and the output $\mathbf{y}$ of such a system by

$$\hat{\mathbf{y}} = \langle \omega, \phi_{\mathbf{x}} \rangle , \tag{15}$$

whereby both the (fixed) element ω and the transformation $\phi_{\mathbf{x}}$ lie on a QRKHS $\mathcal{H}_0$ with inner product $\langle \cdot, \cdot \rangle$. To avoid the need for the optimisation in the high-dimensional space $\mathcal{H}_0$, we restrict the search for the optimal weight to a reduced-dimensional QRKHS $\mathcal{H} = \text{span}\{\phi_{\mathbf{x}_i}, i = 1, \ldots, N\} \subsetneq \mathcal{H}_0$, referred to to as the *empirical feature space* [41]. In this way, the optimal $\omega \in \mathcal{H}$ can be expressed as

$$\omega = \sum_{i=1}^{N} a_i \phi_{\mathbf{x}_i},$$

and therefore the estimate (15) takes the form

$$\hat{\mathbf{y}} = \sum_{i=1}^{N} a_i \left\langle \phi_{\mathbf{x}_i}, \phi_{\mathbf{x}} \right\rangle = \sum_{i=1}^{N} a_i K(\mathbf{x}_i, \mathbf{x}),$$

where K is the generating kernel of the QRKHS $\mathcal{H}$ for which existence and uniqueness is guaranteed by the Riesz representation theorem (Thm. 1). As a consequence, the search for the optimal weights ω is simplified into the problem of finding the coefficients $\mathbf{a} = [a_1, \ldots, a_N] \in \mathbb{H}^N$.

Using the regularised least-squares criterion for finding the optimal vector $\mathbf{a}$ given the observations C_N, and the regularisation parameter $\rho \in \mathbb{R}_+$, we arrive at the optimisation problem

$$\mathbf{a} = \arg\min_{\mathbf{a} \in \mathbb{H}^N} \sum_{j=1}^{N} \left\| \mathbf{y}_j - \sum_{i=1}^{N} a_i K(\mathbf{x}_i, \mathbf{x}_j) \right\|^2 + \rho \|\mathbf{a}\|^2 ,$$

for which the solution can be found in a closed form using the $\mathbb{HR}$ calculus [30], and is given by

$$\mathbf{a} = (\mathbf{K}^H \mathbf{K} + \rho \mathbf{I})^{-1} \mathbf{K}^H \mathbf{Y}, \tag{16}$$

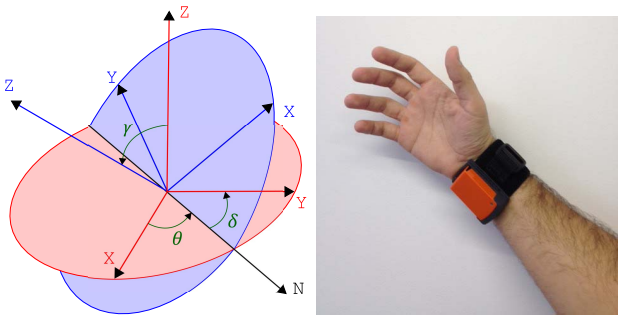

Fig. 3. Inertial body sensor setting. **[Left]** Fixed coordinate system (red), sensor coordinate system (blue) and Euler angles (green). **[Right]** A 3D inertial body sensor at the right wrist.

where $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]^T$, $\mathbf{I}$ is the identity matrix, and $\mathbf{K}$ is the Gram matrix evaluated over the set of training samples given by $\mathbf{K}_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$.

We next present two applications of KRR operating in a QRKHS with the kernels presented in Section V, and refer to the input samples of the training set C_N as *support vectors*. The algorithms were implemented in MATLAB® using the Quaternion Toolbox for MATLAB® [46].

B. Multivariate Body Motion Tracking: Cubic Kernels

We implemented KRR algorithms using real, vector and quaternion cubic kernels to perform a one-step-ahead prediction of the trajectory of limbs in Tai Chi sequences. Data sets for training and validation of the algorithms corresponded to different realisations of Tai Chi movements.

1) Data Acquisition and Preprocessing: Four accelerometers (placed at wrists and ankles) recorded the three Euler angles (Fig. 3), giving a total of 12 signals $\{\theta_s\}_{s=1,..,12}$ taking values in the range $[-\pi, \pi]$. The recorded signals were discontinuous in $\{-\pi, \pi\}$ and thus unsuitable for the application of continuous kernels, hence the angles data were conditioned through the mapping $\theta_s \mapsto (\sin\theta_s, \cos\theta_s)$. These new features also made it possible for the data to be resampled if needed.

Each of the scalar mappings $\theta_s \mapsto \sin\theta_s$, $\theta_s \mapsto \cos\theta_s$ is non-injective (and non-invertible) and therefore does not allow for the angle signal θ_s to be recovered. However, the considered 2D map $\theta_s \mapsto (\sin\theta_s, \cos\theta_s) \in \mathbb{R}^2$ is bijective and therefore invertible, hence, allowing us to recover the original angle signal θ_s. Additionally, the proposed map also allows us to preserve the dynamics of the signal. As illustrated in Fig. 4, sine and cosine preserve data variation successfully only when they behave in a linear-like fashion[5]; however, as such trigonometric functions are shifted versions of one another, by considering them together the signals dynamics are well preserved.

The data corresponding to the so-mapped 12 angles were then represented by a 24-dimensional real signal, or equivalently, a six-dimensional quaternion signal. Therefore, by considering two delayed samples as regressors, the input and output pairs were respectively elements of $\mathbb{H}^{12}$ and $\mathbb{H}^6$.

[5]Recall that for $\theta \approx 0$, $\sin(\theta) \approx \theta$ and $\cos(1.5\pi + \theta) \approx \theta$.

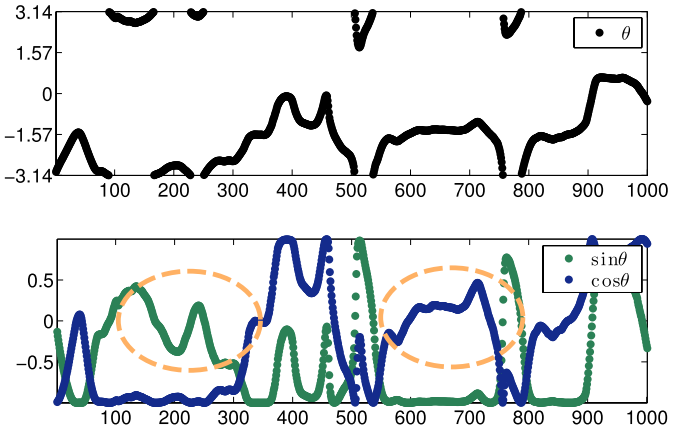

Fig. 4. Raw angle measurements and considered features. **[Top]** Original discontinuous angle recording and **[Bottom]** the corresponding continuous sine and cosine mapping. Observe that in the right (left) circle only cosine (sine) preserves the dynamics of the angle signal accurately.

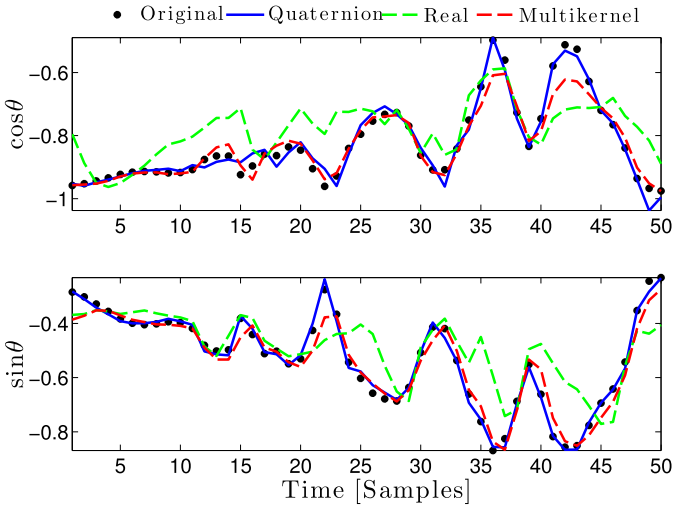

Fig. 5. Body sensor signal tracking: Angle features $(\sin\theta, \cos\theta)$ and KRR estimates.

2) Choice of Cubic Kernels: The kernel ridge regression algorithms were implemented featuring real-, vector-, and quaternion-valued cubic kernels. The real kernel used was the standard cubic kernel in Eq. (10) for $p = 3$, that is,[6] $K_{RP}(\mathbf{x}, \mathbf{y}) = \left(1 + \langle \mathbf{x}, \mathbf{y} \rangle_{\Re}\right)^3$, whereas the vector-kernel chosen was

$$K_M(\mathbf{x}, \mathbf{y}) = \begin{pmatrix} \langle \mathbf{x}, \mathbf{y} \rangle_{\Re}^3 \\ \left(1 + \langle \mathbf{x}, \mathbf{y} \rangle_{\Re}\right)^3 \\ \left(10 + \langle \mathbf{x}, \mathbf{y} \rangle_{\Re}\right)^3 \\ \left(100 + \langle \mathbf{x}, \mathbf{y} \rangle_{\Re}\right)^3 \end{pmatrix}$$

since its subkernels are a basis of the space of cubic polynomials on $\langle \mathbf{x}, \mathbf{y} \rangle_{\Re}$ and performed better that other basis considered (see Appendix A for comments on the choice of these subkernels).

Finally, we chose the quaternion kernel $K_{QP}(\mathbf{x}, \mathbf{y}) = \left(1 + \mathbf{x}^H \mathbf{x}\right)\left(1 + \mathbf{x}^H \mathbf{y}\right)\left(1 + \mathbf{y}^H \mathbf{y}\right)$ introduced in Eq. (11) to validate the quaternion kernel regression concept.

3) Results: We chose a regularisation parameter $\rho = 5$ as this suited all three algorithms and in particular allowed the multikernel not to suffer from overfitting. Fig. 5 shows the cosine and sine of one coordinate θ and their kernel estimates for 90 randomly chosen support vectors. Fig. 6 shows the averaged prediction mean square error (MSE) over 30 independent realisations, as a function of the number of support vectors for the same regularisation parameter. The support vectors and the validation set (50 samples) were randomly chosen, without repetition, for all realisations.

[6]Recall that $\langle \mathbf{x}, \mathbf{y} \rangle_{\Re} = \Re\{\langle \mathbf{x}, \mathbf{y} \rangle\}$.

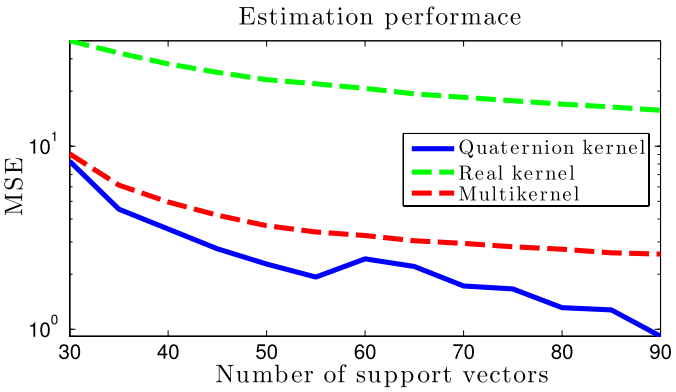

Fig. 6. Performance of KRR algorithms for body sensor signal tracking as a function of the number of support vectors.

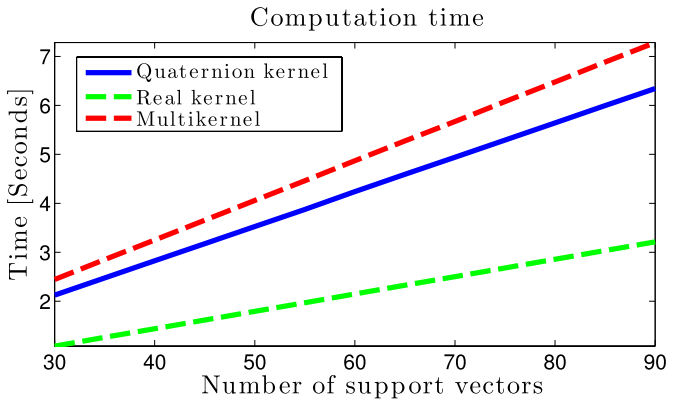

Fig. 7. Computation time of KRR algorithms for body sensor signal tracking.

Observe that the scalar real kernel algorithm is outperformed by both the multikernel and quaternion ones due to their higher degrees of freedom. Moreover, note that the performance of the quaternion cubic kernel became progressively better than that of its real-valued counterpart as the number of support vectors (and therefore training samples) increased. The better performance of K_{QP} for a larger number of support vectors can be explained by the inability of K_{RP} to model cross-coupling between data components and the cross-coordinate terms, for which the quaternion cubic kernel is perfectly well suited (see Remark 9).

Finally, Fig. 7 shows the computation time for all three algorithms. In line with Table I, the complexities of the vector and quaternion kernels were found to be similar and greater than that of the real kernel.

C. Nonlinear Channel Equalisation: Real and Quaternion Gaussian Kernels

We next validated the real and quaternion Gaussian kernels for the problem of nonlinear channel equalisation.

1) Channel Model: The transmission channel was modelled as a linear (moving average) filter with a memoryless nonlinearity stage corrupted by noise:

$$\mathbf{y}_n = a_1\mathbf{x}_n + a_2\mathbf{x}_{n-1}$$
$$\mathbf{s}_n = \mathbf{y}_n + a_3\mathbf{y}_n^2 + \epsilon_n,$$

where $\{\mathbf{x}_n\}_{n\in\mathbb{N}}$ is the transmitted message (input to the channel), $\{\mathbf{y}_n\}_{n\in\mathbb{N}}$ is an unobserved latent process, $\{\epsilon_n\}_{n\in\mathbb{N}}$ is a noise process, and $\{\mathbf{s}_n\}_{n\in\mathbb{N}}$ is the received signal (output of the channel). This model has been previously considered for the validation of kernel learning algorithms including KRR [49], kernel LMS [13], and its complex-valued extensions [16], [17]. The aim of channel equalisation is to identify the original message $\{\mathbf{x}_n\}_{n\in\mathbb{N}}$ from the noisy measurements $\{\mathbf{s}_n\}_{n\in\mathbb{N}}$.

We focused on the quadrivariate case, that is, $\mathbf{x}_n, \mathbf{y}_n, \epsilon_n, \mathbf{s}_n \in \mathbb{R}^4$, and assumed that the components of the input vector (message) $\mathbf{x}_n$ are jointly Gaussian, i.e. $\mathbf{x}_n \sim \mathcal{N}(\mathbf{0}, \Sigma)$, and uncorrelated with the (also Gaussian) noise $\epsilon_n \sim \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{2}})$. The quadriavariate real signals $\mathbf{x}_n, \mathbf{s}_n \in \mathbb{R}^4$ were then expressed as univariate quaternion sequences $x_n, s_n \in \mathbb{H}$.

The model parameters were randomly chosen and had the values

$$\Sigma = \begin{bmatrix} 1.2624 & -0.3541 & -0.1457 & -0.5030 \\ -0.3541 & 0.8487 & -0.1730 & 0.0402 \\ -0.1457 & -0.1730 & 0.4553 & -0.3892 \\ -0.5030 & 0.0402 & -0.3892 & 1.4336 \end{bmatrix},$$
$$a_1 = 0.7466 + i0.3733 - j0.28 + k0.1867$$
$$a_2 = 0.4564 + i0.1521 - j0.6085 + k0.4564$$
$$a_3 = 0.5341 + i0.3204 + j0.1068 - k0.6409.$$

With this choice of parameters, both the original message and the received signals were noncircular quaternion sequences [24], [26].

2) Kernel Parameter Design: Within the KRR setting, once the optimal weights **a** are computed via Eq. (16), the estimate is linear in the kernel evaluations. Accordingly, empirical criteria for kernel design were used to set the kernel parameters so that the kernel evaluations (entries of the kernel evaluation matrix **K**) remained bounded, while at the same time captured enough data variance. We set the parameters of the Gaussian kernels to be $A_R = 6 \cdot 10^{-3}$ (real) and $A_Q = 10^{-4}$ (quaternion) by analysing the second moment of the kernel evaluations over a 200-sample realisation of the process $\mathbf{s}_t$, thus ensuring boundedness and sufficient variability. Fig. 8 analyses the features used for setting kernel parameters and shows the histogram of the kernel evaluations corresponding to the choice of parameters.

3) Validation: The ability of the different kernels to both (i) learn the relationship between the available input-output samples and (ii) generalise the estimates to new datasets of similar dynamics, was next assessed. Both kernels were also compared to the strictly- and widely-linear quaternion ridge regression. See Appendices B and C for an introduction to quaternion widely-linear estimation and the models used for this experiment.

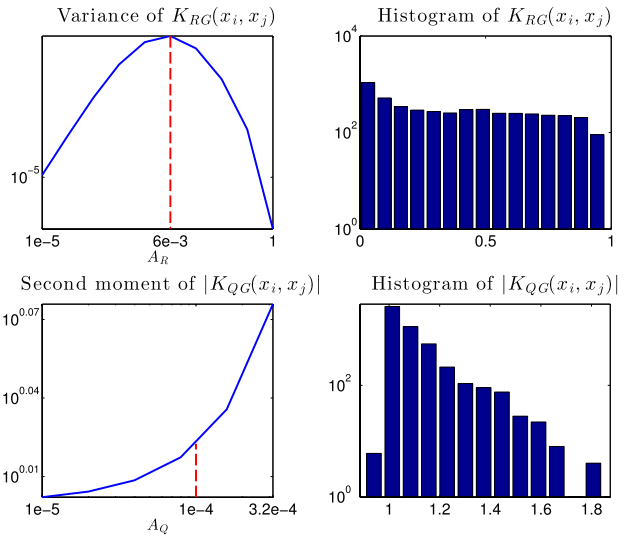

Fig. 8. Choice of kernel parameters (red) A_R and A_Q based on the second moment of the kernel evaluations and histograms corresponding to the chosen parameters.

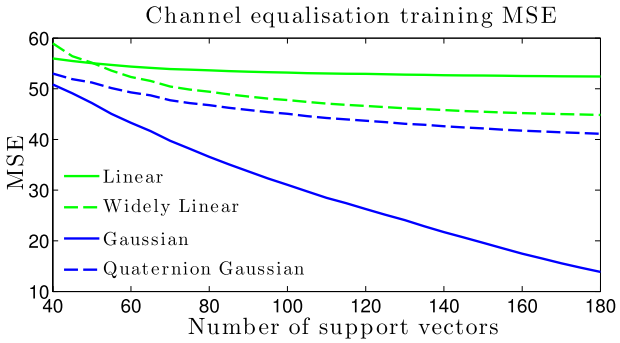

Fig. 9. Training MSE of ridge regression algorithms for channel equalisation.

Fig. 9 shows the training MSE averaged over 30 realisations as a function of the number of support vectors. The training MSE was computed from the estimate of a 200-sample sequence which **contained the support (training) vectors**. Observe that for more than 50 support vectors, the widely-linear ridge regression algorithm outperformed its strictly linear counterpart. Also note that the training performance of the quaternion Gaussian kernel was similar to that of the widely-linear ridge regression algorithm [24]. The real Gaussian KRR offered the best training performance, which improved monotonically with the number of support vectors.

The validation MSE, also averaged over 30 realisations, is shown in Fig. 10 as a function of the number of support vectors. To compute the validation MSE, the support samples (together with the training samples) and the estimated signal **corresponded to different realisations** of 100 samples each, this way, the validation MSE assesses the ability of the regression algorithms to generalise the input-output dependency. Observe that, on average, the quaternion Gaussian kernel provided the best estimates, outperforming not only the linear ridge regression algorithms, but also to the standard, real-valued, Gaussian kernel.

The unbounded nature of the quaternion-valued Gaussian kernel allowed for the extrapolation of the nonlinear behaviour

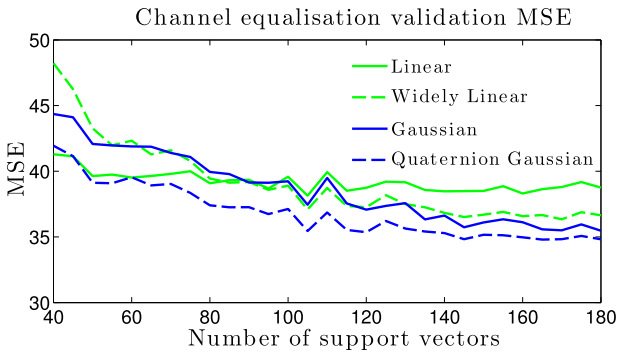

Fig. 10. Validation MSE of ridge regression algorithms for channel equalisation.

learned in the training stage. This property is not found in the real Gaussian kernel, which serves as a similarity measure and is therefore better suited for data interpolation.

Due to the enhanced modelling ability arising from the terms in their imaginary parts, the quaternion-valued kernels were less prone to overfitting than the real-valued kernels, as shown in both experiments. Furthermore, the superior performance of the quaternion SVR approach highlights the advantage of using high-dimensional feature spaces.

VII. Conclusions

We have investigated the existence of quaternion reproducing kernel Hilbert spaces (QRKHS), as well as the advantages of quaternion kernels over vector-kernels. This has been achieved based on (i) a rigorous derivation and analysis of quaternion versions of the Riesz representation and the Moore-Aronszajn theorems, and (ii) an account of the differences between vector and quaternion kernels in terms of both positive definiteness requirements and their associated RKHSs. As a consequence, the design and implementation of kernel estimation algorithms operating on a novel class of high-dimensional quaternion-valued feature spaces has been simplified into the choice of a positive-definite (quaternion) kernel, in a way analogous to that of the real- and complex-valued cases. The improved performance of the quaternion-valued cubic and exponential kernels has been demonstrated in the kernel ridge regression setting, for the 3D body motion tracking and nonlinear channel equalisation applications. The quaternion-valued kernels have been shown to outperform their real- and vector-valued counterparts in the mean square sense, and we have demonstrated their ability to capture the inherent data relationships in a more accurate and physically-meaningful way, while being robust to overfitting. For rigour, the existence and uniqueness results have also been provided with the aim to readily serve as a basis for further quaternion-valued extensions of existing kernel algorithms.

Appendix

A. Basis of Cubic Polynomials in $\langle \mathbf{x}, \mathbf{y} \rangle_{\Re}$

We show that the polynomials $\langle \mathbf{x}, \mathbf{y} \rangle_{\Re}^3$, $\left(1 + \langle \mathbf{x}, \mathbf{y} \rangle_{\Re}\right)^3$, $\left(10 + \langle \mathbf{x}, \mathbf{y} \rangle_{\Re}\right)^3$, $\left(100 + \langle \mathbf{x}, \mathbf{y} \rangle_{\Re}\right)^3$ are a basis of the space of cubic polynomials in $\langle \mathbf{x}, \mathbf{y} \rangle_{\Re}$. For simplicity we denote $\Delta = \langle \mathbf{x}, \mathbf{y} \rangle_{\Re}$.

For $[\alpha, \beta, \gamma, \delta]^T \in \mathbb{R}^4$, we need to find $[a, b, c, d]^T \in \mathbb{R}^4$ such that

$$a\Delta^3 + b\left(1+\Delta\right)^3 + c\left(10+\Delta\right)^3 + d\left(100+\Delta\right)^3 = \alpha\Delta^3 + \beta\Delta^2 + \gamma\Delta + \delta. \quad (17)$$

Upon expanding the left-hand side of Eq. (17) and factorising it with respect to the basis $[\Delta^3, \Delta^2, \Delta, 1]$, we obtain

$$\begin{aligned} &a\Delta^3 + b\left(1+\Delta\right)^3 + c\left(10+\Delta\right)^3 + d\left(100+\Delta\right)^3 \\ &= a\Delta^3 + b\left(1+3\Delta+3\Delta^2+\Delta^3\right) \\ &\quad + c\left(10^3+300\Delta+30\Delta^2+\Delta^3\right) \\ &\quad + d\left(100^3+3\times100^2\Delta+300\Delta^2+\Delta^3\right) \\ &= \Delta^3(a+b+c+d) + \Delta^2(3b+30c+300d) \\ &\quad + \Delta(3b+300c+3\times100^2 d) + 1(b+10^3 c+100^3 d) \quad (18) \end{aligned}$$

A comparison of the right-hand sides of Eqs. (17) and (18) gives the linear equation

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 3 & 30 & 300 \\ 0 & 3 & 300 & 3\times100^2 \\ 0 & 1 & 10^3 & 100^3 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \\ \delta \end{pmatrix}$$

with an invertible matrix on the left-hand side.

As a consequence, the proposed basis is a linearly-independent basis of the set of cubic polynomials in Δ (the independence property can be verified by $[\alpha, \beta, \gamma, \delta] = \mathbf{0} \Rightarrow [a, b, c, d] = \mathbf{0}$).

Other linearly-independent bases of real cubic polynomials on Δ were also considered, including $\{1, \Delta, \Delta^2, \Delta^3\}$ and $\{(b_0+\Delta)^3, (b_1+\Delta)^3, (b_2+\Delta)^3, (b_3+\Delta)^3\}$ for **different** parameters $b_0, b_1, b_2, b_3 \in \mathbb{R}^+$. We have found that the chosen basis (subkernels) provided the best results, in the MSE sense, in the prediction setting considered.

B. Quaternion Widely-Linear Ridge Regression

Strictly-linear models assume that the minimum mean square estimator (MMSE) $E\{\mathbf{x}|\mathbf{s}\}$ of a vector $\mathbf{x}$ given an observation vector $\mathbf{s}$ is, regardless of the real or quaternion nature of the vectors, given by[7]

$$\hat{\mathbf{x}} = \mathbf{A}\mathbf{s}, \quad (19)$$

where $\mathbf{A}$ is a coefficient matrix. On the other hand, widely-linear quaternion models exploit the linear dependency between the vector $\mathbf{x}$ and each of the components of the regressor $\mathbf{s} = \mathbf{s}_r + i\mathbf{s}_i + j\mathbf{s}_j + k\mathbf{s}_k$, yielding an estimator which is linear in each of these components.

Alternatively, by considering the *involutions* of $\mathbf{s}$, given by [50]

$$\begin{aligned} \mathbf{s}^i &= -i\mathbf{s}i = \mathbf{s}_r - i\mathbf{s}_i + j\mathbf{s}_j + k\mathbf{s}_k \\ \mathbf{s}^j &= -j\mathbf{s}j = \mathbf{s}_r + i\mathbf{s}_i - j\mathbf{s}_j + k\mathbf{s}_k \\ \mathbf{s}^k &= -k\mathbf{s}k = \mathbf{s}_r + i\mathbf{s}_i + j\mathbf{s}_j - k\mathbf{s}_k, \end{aligned}$$

we can express the widely-linear estimator in the form [51]:

$$\hat{\mathbf{x}} = \mathbf{A}\mathbf{s} + \mathbf{B}\mathbf{s}^i + \mathbf{C}\mathbf{s}^j + \mathbf{D}\mathbf{s}^k = \mathbf{W}\mathbf{s}^a,$$

[7]We have maintained the notation $\mathbf{s}$ and $\mathbf{x}$ for consistency with the nonlinear channel equalisation simulation.

where

$$\mathbf{W} = [\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}], \quad \mathbf{s}^a = \begin{bmatrix} \mathbf{s} \\ \mathbf{s}^i \\ \mathbf{s}^j \\ \mathbf{s}^k \end{bmatrix}$$

are the so-called *augmented* quantities.

This way, the widely-linear estimator is theoretically equivalent to the quadrivariate real-valued estimator and is the best linear estimator in $\mathbb{H}$ [52]. For complex widely-linear algorithms, see [53], [54].

In the ridge-regression setting, the weights $\mathbf{W}$ are computed in the regularised least-squares sense based on a set of available observation pairs $\{(\mathbf{s}_n, \mathbf{x}_n), n = 1, \ldots, N\}$, that is

$$\mathbf{W} = (\mathbf{S}^H\mathbf{S} + \rho\mathbf{I})^{-1}\mathbf{S}^H\mathbf{X},$$

where $\rho > 0$ is a regularization factor, and $\mathbf{S} = [\mathbf{s}_1, \ldots, \mathbf{s}_N]$ and $\mathbf{X} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_N^T]^T$ are the matrices of available observations.

The widely-linear ridge regression can also be regarded as an approximation of the widely-linear Wiener filter [52], where the correlation matrix and the autocorrelation vector are approximated using the available data.

C. Model Comparison for Nonlinear Channel Equalisation

In Section VI-C we compared the proposed Gaussian kernel (real- and quaternion-valued) algorithms against strictly-linear and widely-linear models. The underlying idea is that a novel nonlinear estimation algorithm is only justified if it outperforms existing (strictly and widely) linear approaches. A compact description of the models implemented in Section VI-C, together with their optimal least-squares parameters, is given as follows

- Strictly linear:

$$\begin{aligned} \hat{\mathbf{x}}_L &= \mathbf{A}\mathbf{s} \\ \mathbf{A} &= (\mathbf{S}^H\mathbf{S} + \rho_1\mathbf{I})^{-1}\mathbf{S}^H\mathbf{X} \\ \mathbf{S} &= [\mathbf{s}_1, \ldots, \mathbf{s}_N], \end{aligned}$$

- Widely linear:

$$\begin{aligned} \hat{\mathbf{x}}_{WL} &= \mathbf{W}\mathbf{s}^a \\ \mathbf{W} &= \left(\mathbf{S}^{aH}\mathbf{S}^a + \rho_2\mathbf{I}\right)^{-1}\mathbf{S}^{aH}\mathbf{X} \\ \mathbf{S}^a &= [\mathbf{s}_1^a, \ldots, \mathbf{s}_N^a], \end{aligned}$$

- Real Gaussian kernel (Eq. (13)):

$$\begin{aligned} \hat{\mathbf{x}}_{RG} &= \sum_{i=1}^{N} a_i K_{RG}(\mathbf{s}_i, \mathbf{s}) \\ [a_1, \ldots, a_N]^T &= (\mathbf{K}^H\mathbf{K} + \rho_3\mathbf{I})^{-1}\mathbf{K}^H\mathbf{X} \\ \mathbf{K}_{i,j} &= K_{RG}(\mathbf{s}_i, \mathbf{s}_j), \end{aligned}$$

- Quaternion Gaussian kernel (Eq. (14)):

$$\begin{aligned} \hat{\mathbf{x}}_{QG} &= \sum_{i=1}^{N} b_i K_{QG}(\mathbf{s}_i, \mathbf{s}) \\ [b_1, \ldots, b_N]^T &= (\mathbf{K}^H\mathbf{K} + \rho_4\mathbf{I})^{-1}\mathbf{K}^H\mathbf{X} \\ \mathbf{K}_{i,j} &= K_{QG}(\mathbf{s}_i, \mathbf{s}_j). \end{aligned}$$

Recall that $\rho_1, \rho_2, \rho_3, \rho_4 > 0$ are regularisation factors, and $\mathbf{X} = [\mathbf{x}_1^T, \ldots, \mathbf{x}_N^T]^T$ is the desired output data.

REFERENCES

[1] V. Vapnik, *Estimation of Dependences Based on Empirical Data*. New York, NY, USA: Springer-Verlag, 1982.

[2] J. Xu, Y. Zhou, and Y. Wang, "A classification of questions using SVM and semantic similarity analysis," in *Proc. 6th Int. Conf. Internet Comput. Sci. Eng.*, Apr. 2012, pp. 31–34.

[3] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Mach. Learn.*, vol. 46, nos. 1–3, pp. 389–422, 2002.

[4] J. H. Min and Y.-C. Lee, "Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters," *Expert Syst. Appl.*, vol. 28, no. 4, pp. 603–614, 2005.

[5] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. 5th Annu. Workshop Comput. Learn. Theory*, 1992, pp. 144–152.

[6] M. A. Aizerman, E. A. Braverman, and L. Rozoner, "Theoretical foundations of the potential function method in pattern recognition learning," *Autom. Remote Control*, vol. 25, pp. 821–837, 1964.

[7] H. Drucker, C. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," in *Neural Information Processing Systems*. Cambridge, MA, USA: MIT Press, 1996, pp. 155–161.

[8] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[9] B. Scholkopf and A. Smola, *Learning With Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2001.

[10] A. Friedman, *Foundations of Modern Analysis*, 2nd ed. New York, NY, USA: Dover, 1982.

[11] N. Aronszajn, "Theory of reproducing kernels," *Trans. Amer. Math. Soc.*, vol. 68, no. 3, pp. 337–404, 1950.

[12] C. Saunders, A. Gammerman, and V. Vovk, "Ridge regression learning algorithm in dual variables," in *Proc. 15th Int. Conf. Mach. Learn.*, 1998, pp. 515–521.

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

[14] C. Richard, J. C. M. Bermudez, and P. Honeine, "Online prediction of time series data with kernels," *IEEE Trans. Signal Process.*, vol. 57, no. 3, pp. 1058–1067, Mar. 2009.

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

[16] P. Bouboulis and S. Theodoridis, "The complex Gaussian kernel LMS algorithm," in *Proc. 20th Int. Conf. Artif. Neural Netw., II*, 2010, pp. 11–20.

[17] P. Bouboulis, S. Theodoridis, and M. Mavroforakis, "The augmented complex kernel LMS," *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4962–4967, Sep. 2012.

[18] F. A. Tobar, A. Kuh, and D. P. Mandic, "A novel augmented complex valued kernel LMS," in *Proc. 7th IEEE Sensor Array Multichannel Signal Process. Workshop*, Jun. 2012, pp. 473–476.

[19] C. Carmeli, E. de Vito, and A. Toigo, "Vector valued reproducing kernel Hilbert spaces of integrable functions and Mercer theorem," *Anal. Appl.*, vol. 4, no. 4, pp. 377–408, Oct. 2006.

[20] M. A. Álvarez, L. Rosasco, and N. D. Lawrence, *Kernels for Vector-Valued Functions*. Hanover, MA, USA: Now Publishers Inc., 2012.

[21] F. Tobar, S.-Y. Kung, and D. P. Mandic, "Multikernel least mean square algorithm," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 2, pp. 265–277, Feb. 2014.

[22] F. A. Tobar and D. P. Mandic, "Multikernel least squares estimation," in *Proc. Sensor Signal Process. Defence Conf.*, Sep. 2012, pp. 1–5.

[23] N. L. Bihan, S. Miron, and J. I. Mars, "MUSIC algorithm for vector-sensors array using biquaternions," *IEEE Trans. Signal Process.*, vol. 55, no. 9, pp. 4523–4533, Sep. 2007.

[24] J. Vía, D. Ramírez, and I. Santamaría, "Properness and widely linear processing of quaternion random vectors," *IEEE Trans. Inf. Theory*, vol. 56, no. 7, pp. 3502–3515, Jul. 2010.

[25] J. Vía, D. P. Palomar, L. Vielva, and I. Santamaria, "Quaternion ICA from second-order statistics," *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1586–1600, Apr. 2011.

[26] C. C. Took and D. P. Mandic, "Augmented second-order statistics of quaternion random signals," *Signal Process.*, vol. 91, no. 2, pp. 214–224, 2011.

[27] J. Navarro-Moreno, R. M. Fernandez-Alcala, and J. C. Ruiz-Molina, "A quaternion widely linear series expansion and its applications," *IEEE Signal Process. Lett.*, vol. 19, no. 12, pp. 868–871, Dec. 2012.

[28] J. Navarro-Moreno, R. M. Fernández-Alcalá, C. C. Took, and D. P. Mandic, "Prediction of wide-sense stationary quaternion random signals," *Signal Process.*, vol. 93, no. 9, pp. 2573–2580, 2013.
[29] W. R. Hamilton, "On quaternions; or on a new system of imaginaries in algebra," *London, Edinburgh Dublin Philosoph. Mag. J. Sci.*, vol. 25, no. 163, pp. 10–13, 1844.
[30] D. P. Mandic, C. Jahanchahi, and C. C. Took, "A quaternion gradient operator and its applications," *IEEE Signal Process. Lett.*, vol. 18, no. 1, pp. 47–50, Jan. 2011.
[31] F. A. Tobar and D. P. Mandic, "The quaternion kernel least squares," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2013, pp. 6128–6132.
[32] T. Ogunfunmi and T. Paul, "An alternative kernel adaptive filtering algorithm for quaternion-valued data," in *Proc. Signal Inform. Process. Assoc. Annu. Summit Conf. (APSIPA)*, Dec. 2012, pp. 1–5.
[33] N. Jacobson, *The Theory of Rings*. Providence, RI, USA: AMS, 1943.
[34] N. Jacobson, *Basic Algebra I*. New York, NY, USA: Dover, 2009.
[35] F. G. Frobenius, "Über lineare substitutionen und bilineare formen," *J. Reine Angew. Math.*, vol. 84, pp. 1–63, 1878.
[36] F. Anderson and K. Fuller, *Rings and Categories of Modules* (Graduate Texts in Mathematics), vol. 13, 2nd ed. New York, NY, USA: Springer-Verlag, 1992.
[37] W. Adkins and S. Weintraub, *Algebra: An Aproach Via Module Theory* (Graduate Texts in Mathematics). New York, NY, USA: Springer-Verlag, 1992.
[38] J. Mercer, "Functions of positive and negative type, and their connection with the theory of integral equations," *Philosoph. Trans. Roy. Soc. London A*, vol. 209, pp. 415–446, Nov. 1909.
[39] G. Pedrick, *Theory of Reproducing Kernels for Hilbert Spaces of Vector Valued Functions* (Studies in Eigenvalue Problems). Lawrence, KS, USA: Univ. Kansas, 1957.
[40] L. Schwartz, "Sous-espaces Hilbertiens d'espaces vectoriels topologiques et noyaux associés (noyaux reproduisants)," *J. d'Anal. Math.*, vol. 13, no. 1, pp. 115–256, 1964.
[41] S.-Y. Kung, *Kernel Methods and Machine Learning*. Cambridge, U.K.: Cambridge Univ. Press, 2014.
[42] M. Gönen and E. Alpaydın, "Multiple kernel learning algorithms," *J. Mach. Learn. Res.*, vol. 12, pp. 2211–2268, Jul. 2011.
[43] M. Yukawa, "Multikernel adaptive filtering," *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4672–4682, Sep. 2012.
[44] M. Yukawa, "Nonlinear adaptive filtering techniques with multiple kernels," in *Proc. Eur. Signal Process. Conf.*, 2011, pp. 136–140.
[45] J. P. Ward, *Quaternions and Cayley Numbers: Algebra and Applications*. Norwell, MA, USA: Kluwer, 1997.
[46] S. J. Sangwine and N. L. Bihan. (2005). *Quaternion Toolbox for MATLAB* [Online]. Available: http://qtfm.sourceforge.net/
[47] I. Steinwart, D. Hush, and C. Scovel, "An explicit description of the reproducing kernel Hilbert spaces of Gaussian RBF kernels," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4635–4643, Oct. 2006.
[48] F. A. Tobar, L. Yacher, R. Paredes, and M. E. Orchard, "Anomaly detection in power generation plants using similarity-based modeling and multivariate analysis," in *Proc. Amer. Control Conf. (ACC)*, Jun./Jul. 2011, pp. 1940–1945.
[49] C.-J. Lin, S.-J. Hong, and C.-Y. Lee, "Using least squares support vector machines for adaptive communication channel equalization," *Int. J. Appl. Sci. Eng.*, vol. 3, no. 1, pp. 51–59, 2005.
[50] T. A. Ell and S. J. Sangwine, "Quaternion involutions and anti-involutions," *Comput. Math. Appl.*, vol. 53, no. 1, pp. 137–143, 2007.
[51] C. C. Took and D. P. Mandic, "A quaternion widely linear adaptive filter," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4427–4431, Aug. 2010.
[52] C. Jahanchahi, C. C. Took, and D. P. Mandic, "On HR calculus, quaternion valued stochastic gradient, and adaptive three dimensional wind forecasting," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2010, pp. 1–5.
[53] S. C. Douglas, "Widely-linear recursive least-squares algorithm for adaptive beamforming," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process., (ICASSP)*, Apr. 2009, pp. 2041–2044.
[54] D. P. Mandic and S. L. Goh, *Complex Valued Nonlinear Adaptive Filters: Noncircularity, Widely Linear and Neural Models*. New York, NY, USA: Wiley, 2009.

Felipe A. Tobar received the B.Sc. and M.Sc. degrees in Electrical and Electronic Engineering from Universidad de Chile, Santiago, Chile in 2008 and 2010 respectively. He is currently pursuing his Ph.D. degree in Signal Processing at Imperial College London, London, U.K.

His research interests lie within the interface between signal processing and machine learning, and include high-dimensional kernel regression, Bayesian system identification, and nonlinear adaptive filtering.

Danilo P. Mandic is a Professor in signal processing with Imperial College London, London, U.K., where he has been working in the area of nonlinear adaptive signal processing and nonlinear dynamics. He has been a Guest Professor with Katholieke Universiteit Leuven, Leuven, Belgium and a Frontier Researcher with RIKEN, Tokyo. His publication record includes two research monographs titled *Recurrent Neural Networks for Prediction* (West Sussex, U.K.: Wiley, August 2001) and *Complex-Valued Nonlinear Adaptive Filters: Noncircularity, Widely Linear and Neural Models* (West Sussex, U.K.: Wiley, April 2009), an edited book titled *Signal Processing for Information Fusion* (New York: Springer, 2008), and more than 200 publications on signal and image processing.

He has been a member of the IEEE Technical Committees on Signal Processing Theory and Methods and Machine Learning for Signal Processing, an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE TRANSACTIONS ON NEURAL NETWORKS, and the IEEE SIGNAL PROCESSING MAGAZINE. He has produced award winning papers and products from his collaboration with the industry, and has received President's Award for excellence in postgraduate supervision at Imperial College. He is a member of the London Mathematical Society.