Design of Positive-Definite Quaternion Kernels

Felipe Tobar and Danilo P. Mandic

Abstract—**Quaternion reproducing kernel Hilbert spaces (QRKHS) have been proposed recently and provide a high-dimensional feature space (alternative to the real-valued multikernel approach) for general kernel-learning applications. The current challenge within quaternion-kernel learning is the lack of general quaternion-valued kernels, which are necessary to exploit the full advantages of the QRKHS theory in real-world problems. This letter proposes a novel way to design quaternion-valued kernels, this is achieved by transforming three complex kernels into quaternion ones and then combining their real and imaginary parts. Building on this general construction, our emphasis is on a new quaternion kernel of polynomial features, which is assessed in the prediction of bodysensor networks applications.**

Index Terms—**Quaternion kernels, complex kernels, multiple kernels, vector kernels.**

I. INTRODUCTION

Kernel learning algorithms have become a standard in both classification and regression problems, and have found application in econometrics, tracking, robotics, and fault diagnosis. The two main aspects of kernel estimation are the *learning algorithm* and the *choice of the kernel*. The learning algorithm is usually obtained through the optimisation of a cost function chosen to best suit the problem at hand; in kernel regression, for example, the kernel weights can be found through kernel least mean square [1], [2], kernel recursive least squares [3], and kernel ridge regression [4]. While kernel learning algorithms are already maturing, the choice of the kernel and its effect on the performance of the estimation algorithm still remains an open issue. The kernel design can be addressed by a mixture of real-valued kernels, [5], [6], while a more recent alternative is to consider complex-valued [7], [8] and quaternion-valued [9], [10] kernels.

Recent advances in sensor technology have provided access to 3D data, this has spurred a resurgence in research on quaternion representations of such signals. Indeed, quaternions have become a standard in a number of areas, including computer graphics, quantum physics and aeronautics. When it comes to quaternion kernel algorithms, theoretical foundations for the existence of quaternion RKHS (QRKHS) and the validity of quaternion versions of established real-valued kernel algorithms have been provided in [9]. However, quaternion extensions of standard kernels are limited to only the linear and Gaussian ones. This is a consequence of the

F. Tobar is with the Department of Engineering, University of Cambridge, UK (e-mail: fat25@cam.ac.uk), and acknowledges financial support to EPSRC grant number EP/L000776/1.

D. P. Mandic is with the Department of Electrical and Electronic Engineering, Imperial College London, UK.

algebraic properties of the quaternion ring, in particular, the non-commutativity of its product. This implies that popular kernels which are positive definite in the real or complex fields, cannot be straightforwardly extended to the quaternion ring (e.g. the polynomial kernel) as their positive definiteness relies upon the assumption of commutativity, thus limiting the range of applications.

Quaternion-valued kernels (and not just kernels that admit quaternion-valued inputs) are appealing, since their feature mapping is also quaternion-valued and corresponds to four real-valued maps – or two complex-valued ones. As a consequence, the quaternion feature space (a quaternion RKHS), which is the span of the feature mapping, is of higher dimensionality than standard real/complex RKHSs and, therefore, can be interpreted as a *richer feature space* [11].

Along these lines, the aim of this paper is threefold: (i) to propose a novel way of constructing Hermitian, positive semi-definite (PSD), quaternion kernels from three complex-valued ones, this establishes a sufficient condition for the Hermitian and PSD properties of the constructed quaternion-valued kernel; (ii) to introduce a polynomial quaternion-kernel, thus bridging a gap in the open literature; and (iii) to validate the proposed quaternion polynomial (cubic) kernel against existing approaches on a 3D body-sensor application.

II. QUATERNIONS: BACKGROUND AND NOTATION

The main difference between the algebraic properties of the quaternion ring and those of the real/complex field is the lack of the commutativity property. In this section, we state the properties used throughout the paper and provide an example of a kernel that is PSD in the real and complex fields but not in the quaternion ring — for a detailed account see [9].

A. Definitions and Identities

Let $\{1, i, j, k\}$ be the quaternion basis, $q = q_r + iq_i + jq_j + kq_k \in \mathbb{H}$ a quaternion scalar and $\mathbf{P} \in \mathbb{H}^{n \times m}$, $\mathbf{Q} \in \mathbb{H}^{m \times n}$ quaternion matrices. We will use the following operations:

- Conjugate operator: $q^* = q_r - iq_i - jq_j - kq_k$,
- Hermitian operator: $\mathbf{P}^H = (\mathbf{P}^*)^T$, where the conjugate operator is applied element-wise, i.e., $\{\mathbf{P}^*\}_{r,s} = \{\mathbf{P}\}_{r,s}^*$,
- Hermitian of product: $(\mathbf{PQ})^H = \mathbf{Q}^H \mathbf{P}^H$,
- Imaginary part: $\Im \mathbf{P} = \frac{1}{2}(\mathbf{P} - \mathbf{P}^*)$,
- Real part: $\Re \mathbf{P} = \frac{1}{2}(\mathbf{P} + \mathbf{P}^*)$,
- The square quaternion matrix $\mathbf{P} \in \mathbb{H}^{n \times n}$ is positive semidefinite iff $\mathbf{x}^H \mathbf{P} \mathbf{x} \geq 0, \forall \mathbf{x} \in \mathbb{H}^n$.

A quaternion-valued kernel is a two-input function $K(\cdot, \cdot)$, $K : X \times X \to \mathbb{H}$, where X is an arbitrary sample set. The transpose of a kernel is defined as $K(\mathbf{x}, \mathbf{y})^T = K(\mathbf{y}, \mathbf{x})$ and we say that the kernel is symmetric if and only if $K(\mathbf{x}, \mathbf{y}) = K(\mathbf{y}, \mathbf{x}) \ \forall \mathbf{x}, \mathbf{y} \in X$. We can now introduce Mercer kernels.

Definition 1 (quaternion-Mercer kernel [12], [13])**.** *A quaternion kernel K is a quaternion-Mercer kernel iff it is:*

(i) *Hermitian (or equal to its conjugate transpose) i.e., $K(\mathbf{x}, \mathbf{y}) = K^H(\mathbf{x}, \mathbf{y}) = (K^T(\mathbf{x}, \mathbf{y}))^* = K^*(\mathbf{y}, \mathbf{x})$, and*
(ii) *Positive semidefinite on X, i.e., the Gram matrix defined element-wise by $\mathbf{K}_{r,s} = K(\mathbf{s}_r, \mathbf{s}_s) \in \mathbb{H}$ is positive semidefinite for any set $S = \{\mathbf{s}_1, \ldots, \mathbf{s}_m\} \subset X, m \in \mathbb{N}$.*

Notice that we denote **functions** $K : X \times X \to \mathbb{H}$ by uppercase K, whereas **matrices** comprising values of K are denoted using the bold font $\mathbf{K}$ - subindices will also be used for clarity when referring to more than one kernel function. Additionally, by replacing $\mathbb{H}$ by $\mathbb{R}$ or $\mathbb{C}$ in Def. 1, we refer to real-Mercer, complex-Mercer and quaternion-Mercer kernels.

B. The Standard Polynomial Kernel is not Positive Semidefinite in the Quaternion Domain

As an illustrative example, consider the polynomial kernel. Some real-world applications require highly-nonlinear estimation; this can be achieved by using products (or *monomials*) of the entries of the input vector $\mathbf{x}$ as features. In the real- and complex-valued cases, polynomial classifiers [14] employ the feature space of all monomials of order up to p. The resulting feature space is an RKHS, the reproducing kernel of which is the polynomial kernel

$$K_{c,p} = \left(c + \mathbf{x}^H \mathbf{y}\right)^p, \ c \geq 0, p \in \mathbb{N}. \tag{1}$$

The expansion of the polynomial kernel, required to reveal the monomials of order up to p (i.e. the features), requires commutativity of products of the monomials (see [15] for the case $p = 2$). However, as quaternions do not commute, the kernel in eq. (1) is suitable for only real and complex inputs.

To show that the polynomial kernel $K_{c,p}$ in eq. (1) is not necessarily PSD in the quaternion domain, consider the quadratic homogeneous case with scalar input, that is, $c = 0$ and $p = 2$. The kernel in eq. (1) then takes the form

$$K_{0,2} = (x^* y)^2, \ x, y \in \mathbb{H} \tag{2}$$

where, according to Definition 1, a sufficient condition to reject the PSD-ness of $K_{0,2}$ is to find a set S such that the Gram matrix of $K_{0,2}$ constructed from the samples in $S \subset \mathbb{H}$ is not PSD. We proceed by considering the set $S = \{i, j, k\}$, for which the Gram matrix of $K_{0,2}$ is given by

$$\mathbf{K} = \left[\begin{array}{ccc} K_{0,2}(i,i) & K_{0,2}(i,j) & K_{0,2}(i,k) \\ K_{0,2}(j,i) & K_{0,2}(j,j) & K_{0,2}(j,k) \\ K_{0,2}(k,i) & K_{0,2}(k,j) & K_{0,2}(k,k) \end{array} \right] = \left[\begin{array}{ccc} 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{array} \right]$$

The above matrix is not PSD, since its eigenvalues are $\lambda_1 = -1, \lambda_2 = 2, \lambda_3 = 2$, hence, according to Definition 1, the polynomial kernel $K_{c,p}$ is in general not PSD in $\mathbb{H}$.

This example suggests that extending standard real/complex-valued kernels to operate in the quaternion domain is not straightforward. To this end, we next introduce a procedure to design quaternion kernels based on three arbitrary complex kernels.
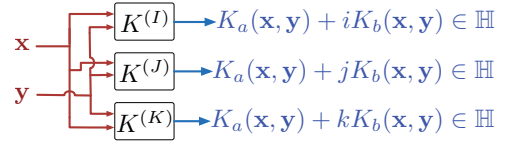

Fig. 1: Quaternion-Mercer kernels $K^{(I)}$, $K^{(J)}$ and $K^{(K)}$. Red lines denote arbitrary inputs and the blue ones quaternion outputs.

III. DESIGN OF QUATERNION KERNELS FROM COMPLEX KERNELS

Our aim is to design quaternion-Mercer kernels (see Definition 1); in the build-up of our main result in Theorem 1, the following lemma establishes a link between complex- and quaternion-Mercer kernels.

Lemma 1. *If $K : X \times X \to \mathbb{C}$ is a **complex**-Mercer kernel with real and imaginary parts respectively given by the functions $K_a, K_b : X \times X \to \mathbb{R}$, then the following three kernels $K^{(I)}, K^{(J)}, K^{(K)} : X \times X \to \mathbb{H}$ defined by*

$$K^{(I)} = K_a + iK_b, \ K^{(J)} = K_a + jK_b, \ K^{(K)} = K_a + kK_b \tag{3}$$

*are **quaternion**-Mercer kernels.*

Proof. Recall that, according to Def. 1, the complex-Mercer kernel $K_a + iK_b$ is Hermitian and PSD. We first prove the Hermitian and PSD properties for $K^{(I)}$:
(i) Since $K_a + iK_b = (K_a + iK_b)^H$ is (complex) Hermitian, then the function $K_a(\mathbf{x}, \mathbf{y}) = K_a^T(\mathbf{x}, \mathbf{y})$ is symmetric and the function $K_b(\mathbf{x}, \mathbf{y}) = -K_b^T(\mathbf{x}, \mathbf{y})$ is antisymmetric (or skew-symmetric) . Therefore, $\left(K^{(I)}\right)^H = K_a^H - iK_b^H = K_a + iK_b$ is Hermitian in $\mathbb{H}$.
(ii) The PSD-ness of $K^{(I)}$ can be expressed in terms of its Gram matrix as $\mathbf{x}^H \mathbf{K}^{(I)} \mathbf{x} \geq 0, \ \forall \mathbf{x} \in \mathbb{H}^n$, or equivalently,

$$\Re\{\mathbf{x}^H \mathbf{K}^{(I)} \mathbf{x}\} \geq 0 \text{ and } \Im\{\mathbf{x}^H \mathbf{K}^{(I)} \mathbf{x}\} = 0. \tag{4}$$

Observe, however, that the right-hand-side identity above always holds for Hermitian kernels:

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

Upon denoting $\mathbf{x} = \mathbf{x}_r + i\mathbf{x}_i + j\mathbf{x}_j + k\mathbf{x}_k \in \mathbb{H}^n$ and expanding the l.h.s of eq. (4), we obtain

$$\Re\{\mathbf{x}^H \mathbf{K}^{(I)} \mathbf{x}\} = \left(\mathbf{x}_r^T \mathbf{K}_a \mathbf{x}_r + \mathbf{x}_i^T \mathbf{K}_a \mathbf{x}_i - 2\mathbf{x}_r^T \mathbf{K}_b \mathbf{x}_i\right) + \left(\mathbf{x}_j^T \mathbf{K}_a \mathbf{x}_j + \mathbf{x}_k^T \mathbf{K}_a \mathbf{x}_k - 2\mathbf{x}_j^T \mathbf{K}_b \mathbf{x}_k\right)$$

where each one of the terms in the parentheses is greater or equal than zero (as they are PSD quadratic forms); consequently, $\Re\{\mathbf{x}^H \mathbf{K}^{(I)} \mathbf{x}\} \geq 0$.
The Hermitian and PSD conditions for $K^{(J)}$ and $K^{(K)}$ follow from the permutation of the indices $\{i, j, k\}$. This concludes the proof. $\square$

In the sequel, we adopt the notation $K^{(I)}, K^{(J)}, K^{(K)}$ to denote the quaternion kernels generated from the complex kernel K as in eq. (3), see also fig. 1 for a block diagram. Observe that these kernels are quaternion-valued with **only one** nonzero imaginary part (i-imaginary, j-imaginary and k-imaginary, respectively). We now state our main result.

Theorem 1. *For an arbitrary triplet of complex-Mercer kernels $K_1, K_2, K_3 : X \times X \to \mathbb{C}$ defined by*

$$K_1 = K_{a_1} + iK_{b_1}, \ K_2 = K_{a_2} + iK_{b_2}, \ K_3 = K_{a_3} + iK_{b_3} \quad (5)$$

where $K_{a_1}, K_{b_1}, K_{a_2}, K_{b_2}, K_{a_3}, K_{b_3} : X \times X \to \mathbb{R}$, the expression

$$\begin{aligned} K(\mathbf{x}, \mathbf{y}) = {} & K_{a_1}(\mathbf{x}, \mathbf{y}) + K_{a_2}(\mathbf{x}, \mathbf{y}) + K_{a_3}(\mathbf{x}, \mathbf{y}) \quad (6) \\ & + iK_{b_1}(\mathbf{x}, \mathbf{y}) + jK_{b_2}(\mathbf{x}, \mathbf{y}) + kK_{b_3}(\mathbf{x}, \mathbf{y}) \end{aligned}$$

is a quaternion-Mercer kernel. Using the notation $K_1^{(I)}, K_2^{(J)}, K_3^{(K)}$, we can express the so-constructed quaternion kernel K in compact form as

$$K(\mathbf{x}, \mathbf{y}) = K_1^{(I)}(\mathbf{x}, \mathbf{y}) + K_2^{(J)}(\mathbf{x}, \mathbf{y}) + K_3^{(K)}(\mathbf{x}, \mathbf{y}). \quad (7)$$

Proof. For the complex-Mercer kernels K_1, K_2, K_3 in eq. (5), Lemma 1 guarantees that the kernels $K_1^{(I)}, K_2^{(J)}, K_3^{(K)}$ are quaternion-Mercer; therefore, the proof follows from the fact that Hermitian and PD functions are closed under addition. □

Notice that existence of the three complex-Mercer kernels K_1, K_2 and K_3 is a sufficient, but not necessary, condition for the existence of the quaternion-Mercer kernel K; this means that not all quaternion kernels are necessarily constructed in such a way. Furthermore, observe that Theorem 1 guarantees the existence of quaternion kernels over any set X where complex kernels can be defined.

IV. A QUATERNION POLYNOMIAL KERNEL

With Theorem 1 we are now equipped to construct a quaternion polynomial kernel using complex polynomial kernels. We proceed by (i) constructing the kernel based on Theorem 1, (ii) accommodating the complex kernels to admit quaternion inputs, and (iii) showing that the constructed kernel has polynomial features and quaternion-valued coefficients.

A. Construction of the Kernel

Let us denote the complex p^{th}-order polynomial kernel by[1] $K_c = (c + \mathbf{x}^H \mathbf{y})^p, \mathbf{x}, \mathbf{y} \in \mathbb{C}^n, c > 0, p \in \mathbb{N}$. By virtue of Theorem 1, the expression

$$K(\mathbf{x}, \mathbf{y}) = K_{c_1}^{(I)}(\mathbf{x}, \mathbf{y}) + K_{c_2}^{(J)}(\mathbf{x}, \mathbf{y}) + K_{c_3}^{(K)}(\mathbf{x}, \mathbf{y}) \quad (8)$$

is a quaternion-Mercer kernel. However, K is defined for complex-valued inputs and therefore cannot be used for regression between quaternion vector spaces unless a transformation of the input is considered.

B. Complexification of the Input

The kernel in eq. (8) can be modified to admit a quaternion-valued (or any vector-valued) input via a *complexification* [16] procedure, that is, by mapping the input set X onto a complex-valued vector space. We consider three different complexification maps, one for each kernel in the r.h.s. of
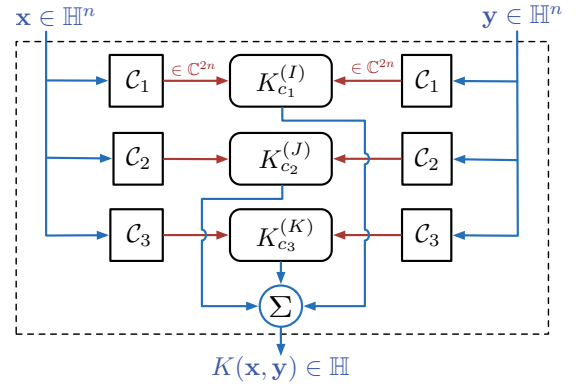
Fig. 2: Quaternion-Mercer kernel constructed from complex-Mercer kernels via complexification. Blue lines denote quaternions, red lines complex numbers, and the dashed line the block $(\mathbf{x}, \mathbf{y}) \to K(\mathbf{x}, \mathbf{y})$.

eq. (8). For the input $\mathbf{x} = \mathbf{x}_r + i\mathbf{x}_i + j\mathbf{x}_j + k\mathbf{x}_k \in \mathbb{H}^n$ the three complexification mappings considered are (see fig. 2)

$$\mathcal{C}_1 : \mathbf{x} \in \mathbb{H}^n \mapsto [\mathbf{x}_r^T \ \mathbf{x}_k^T]^T + i[\mathbf{x}_j^T \ \mathbf{x}_i^T]^T \in \mathbb{C}^{2n} \quad (9)$$

$$\mathcal{C}_2 : \mathbf{x} \in \mathbb{H}^n \mapsto [\mathbf{x}_r^T \ \mathbf{x}_j^T]^T + i[\mathbf{x}_i^T \ \mathbf{x}_k^T]^T \in \mathbb{C}^{2n} \quad (10)$$

$$\mathcal{C}_3 : \mathbf{x} \in \mathbb{H}^n \mapsto [\mathbf{x}_r^T \ \mathbf{x}_i^T]^T + i[\mathbf{x}_k^T \ \mathbf{x}_j^T]^T \in \mathbb{C}^{2n}. \quad (11)$$

The complexification allows for the quaternion-Mercer kernel in eq. (8) to be a function defined between quaternion-valued spaces and is part of the model design.

C. Polynomial Features and Quaternion Coefficients

The proposed polynomial kernel in eq. (8) with the complexification maps in eqs. (9)-(11) can be expressed as a function $K : \mathbb{H}^n \times \mathbb{H}^n \mapsto \mathbb{H}$ given by

$$K(\mathbf{x}, \mathbf{y}) = \quad (12)$$
$$= K_{c_1}^{(I)}(\mathcal{C}_1\mathbf{x}, \mathcal{C}_1\mathbf{y}) + K_{c_2}^{(J)}(\mathcal{C}_2\mathbf{x}, \mathcal{C}_2\mathbf{y}) + K_{c_3}^{(K)}(\mathcal{C}_3\mathbf{x}, \mathcal{C}_3\mathbf{y}).$$

In order to analyse the feature space corresponding to K, let us expand the first term above to give

$$K_{c_1}^{(I)}(\mathcal{C}_1\mathbf{x}, \mathcal{C}_1\mathbf{y}) =$$
$$= \big(c_1 + (\mathbf{x}_r + i\mathbf{x}_j)^H(\mathbf{y}_r + i\mathbf{y}_j) + (\mathbf{x}_i + i\mathbf{x}_k)^H(\mathbf{y}_i + i\mathbf{y}_k)\big)^p$$

which can be written as a quadratic form of the **real vectors** $[\mathbf{x}_r^T \ \mathbf{x}_i^T \ \mathbf{x}_j^T \ \mathbf{x}_k^T]^T, [\mathbf{y}_r^T \ \mathbf{y}_i^T \ \mathbf{y}_j^T \ \mathbf{y}_k^T]^T \in \mathbb{R}^{4n}$ exponentiated to p. Similarly for the remaining two terms, the quaternion-Mercer kernel in eq. (12) can be expressed as

$$K(\mathbf{x}, \mathbf{y}) = \quad (13)$$
$$= \big(c_1 + [\mathbf{x}_r^T \ \mathbf{x}_i^T \ \mathbf{x}_j^T \ \mathbf{x}_k^T]\mathbf{M}_i[\mathbf{y}_r^T \ \mathbf{y}_i^T \ \mathbf{y}_j^T \ \mathbf{y}_k^T]^T\big)^p$$
$$+ \big(c_2 + [\mathbf{x}_r^T \ \mathbf{x}_i^T \ \mathbf{x}_j^T \ \mathbf{x}_k^T]\mathbf{M}_j[\mathbf{y}_r^T \ \mathbf{y}_i^T \ \mathbf{y}_j^T \ \mathbf{y}_k^T]^T\big)^p$$
$$+ \big(c_3 + [\mathbf{x}_r^T \ \mathbf{x}_i^T \ \mathbf{x}_j^T \ \mathbf{x}_k^T]\mathbf{M}_k[\mathbf{y}_r^T \ \mathbf{y}_i^T \ \mathbf{y}_j^T \ \mathbf{y}_k^T]^T\big)^p$$

where the coefficient matrices $\mathbf{M}_i, \mathbf{M}_j, \mathbf{M}_k \in \mathbb{H}^{4n \times 4n}$ are Toeplitz and respectively given in block-wise form as

$$\begin{bmatrix} \mathbf{I} & \mathbf{0} & i\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & i\mathbf{I} \\ -i\mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -i\mathbf{I} & \mathbf{0} & \mathbf{I} \end{bmatrix}, \begin{bmatrix} \mathbf{I} & -j\mathbf{I} & \mathbf{0} & \mathbf{0} \\ j\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & j\mathbf{I} \\ \mathbf{0} & \mathbf{0} & -j\mathbf{I} & \mathbf{I} \end{bmatrix}, \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & k\mathbf{I} \\ \mathbf{0} & \mathbf{I} & k\mathbf{I} & \mathbf{0} \\ \mathbf{0} & -k\mathbf{I} & \mathbf{I} & \mathbf{0} \\ -k\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}$$

where $\mathbf{I}, \mathbf{0} \in \mathbb{H}^{n \times n}$ are the identity and zero matrices.

The proposed kernel in eq. (13) is a sum of three polynomials of the real and imaginary parts of the input $\mathbf{x}$, with quaternion-valued coefficients. Furthermore, notice that although the matrix $\mathbf{M}_i$ (*cf.* $\mathbf{M}_j, \mathbf{M}_k$) depends on the choice of complexification map, it only has real and i−imaginary (*cf.* j−imaginary, k−imaginary) components for any choice of complexification. As a consequence, each of the terms in the parentheses in eq. (13) features a real part and a single imaginary part, in other words, the kernel does not include products of imaginary units, which leads to the lack of PSD-ness as is the case for the polynomial kernel in eq. (2).

The proposed kernel in eq. (13) is indeed a quaternion-Mercer polynomial kernel, since (i) it is a polynomial of the scalar entries of $\mathbf{x}$, (ii) has quaternion-valued coefficients, and (iii) is positive definite by construction (Theorem 1).

V. SIMULATIONS

The proposed polynomial quaternion-Mercer kernel was validated for the cubic case (i.e. $p = 3$) against three existing approaches for the prediction of inertial body motion data.

A. Data, Learning Algorithm and Kernels Considered

The data were obtained using four inertial body sensors, placed on wrists and ankles of an athlete performing Tai-Chi sequences, which recorded 3D angular acceleration. Each angular signal was then mapped according to $\theta_i \to (\cos(\theta_i), \sin(\theta_i))$ to avoid discontinuity close to $\theta_i = \pm\pi$ [9]. The resulting time series was 24-dimensional; in a one-step-ahead prediction setting with two delayed observations, the regression problem required to learn a mapping $\mathbb{R}^{48} \mapsto \mathbb{R}^{24}$, or equivalently, $\mathbb{H}^{12} \mapsto \mathbb{H}^{6}$.

The training and validation trials corresponded to different Tai-Chi sequences, and four kernel ridge regression (KRR, with regularisation factor $\rho = 0.1$) estimators were implemented using the following cubic kernels:
Proposed quaternion kernel: The polynomial quaternion-Mercer kernel in eq. (13), constructed using complex-valued cubic kernels.
Naïve quaternion: The quaternion cubic kernel proposed in [9] given by $K_N(\mathbf{x}, \mathbf{y}) = (1 + \mathbf{x}^H\mathbf{x})(1 + \mathbf{x}^H\mathbf{y})(1 + \mathbf{y}^H\mathbf{y})$. Although this kernel has cubic growth, observe that it is essentially a linear kernel multiplied by the square norm of the inputs $\mathbf{x}$ and $\mathbf{y}$.
Real: The (real) cubic kernel $K_r(\mathbf{x}, \mathbf{y}) = \left(1 + \Re\{\mathbf{x}^H\mathbf{y}\}\right)^3$.
Multikernel: This kernel comprises four real-valued polynomial sub-kernels given by $\left[(\mathbf{x}^H\mathbf{y})^3, (c_1 + \mathbf{x}^H\mathbf{y})^3, (c_2 + \mathbf{x}^H\mathbf{y})^3, (c_3 + \mathbf{x}^H\mathbf{y})^3\right]$, which are a basis for cubic polynomials [5], [9].

The last three kernels were considered in [9] using the same Tai-Chi data, where the naïve kernel outperformed the other two as the number of support vectors increased.

B. Results

Fig. 3 shows the performance of the algorithms considered, evaluated through the normalised mean-square error (top) and computational complexity (bottom). Recall that all four
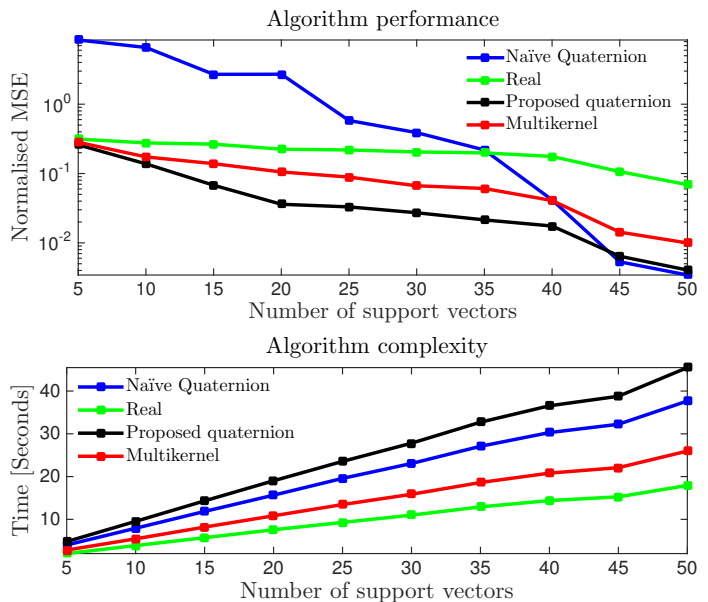

Fig. 3: Performance (top) and computational complexity (bottom) of the quaternion kernel algorithms considered, evaluated for the one-step-ahead prediction of body-motion trajectories.

kernels considered are polynomial and will behave similarly for increasing numbers of support vectors, however, the proposed quaternion kernel outperformed the other kernels for fewer support vectors. This is a consequence of the higher dimensionality of the quaternion-valued feature space, which provides richer features for fewer support vectors.

The higher computational complexity of the proposed kernel is still on par with the other kernels considered, and given by the number of inner products required (three, similarly to the naïve quaternion kernel). Notice that the multikernel algorithm requires four inner products and is, in general, more expensive; however, in the above experiment these inner products were identical and computed only once, thus reducing the complexity of the multikernel algorithm.

VI. CONCLUSIONS

We have proposed a principled approach to construct Hermitian, positive semi-definite, quaternion kernels from complex kernels. This opens completely new possibilities for the implementation of quaternion-kernel algorithms by leveraging upon readily available complex kernels. Building on this theoretical result, we have also proposed a quaternion polynomial kernel, and have demonstrated its usefulness in the prediction of 3D body-motion trajectories. Owing to its higher dimensionality, the proposed quaternion polynomial kernel has been shown to outperform the existing kernels of similar computational complexity. Finally, hypercomplex kernels for signal estimation are in fast development and there is still much room for contributions; for instance, recall from Theorem 1 that the proposed construction is sufficient but not necessary (i.e. not all quaternion kernels can be constructed from three complex kernels), therefore, other ways of constructing quaternion kernels are yet to be explored, e.g. using real kernels or Cayley-Dickson representations.

REFERENCES

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

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

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

[4] H. Drucker, C. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," in *Proc. Advances in Neural Information Processing Systems 9*, 1996, pp. 155–161.

[5] F. Tobar, S.-Y. Kung, and D. Mandic, "Multikernel least mean square algorithm," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 25, no. 2, pp. 265–277, 2014.

[6] M. Gönen and E. Alpaydin, "Multiple kernel learning algorithms," *The Journal of Machine Learning Research*, vol. 12, pp. 2211–2268, 2011.

[7] F. Tobar, A. Kuh, and D. P. Mandic, "A novel augmented complex valued kernel LMS," in *Proc. of the 7th IEEE Sensor Array and Multichannel Signal Processing Workshop*, 2012, pp. 481–484.

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

[9] F. Tobar and D. Mandic, "Quaternion reproducing kernel Hilbert spaces: Existence and uniqueness conditions," *IEEE Transactions on Information Theory*, vol. 60, no. 9, pp. 5736–5749, Sept 2014.

[10] T. Paul and T. Ogunfunmi, "A kernel adaptive algorithm for quaternion-valued inputs," *IEEE Trans. on Neural Networks and Learning Systems,*, vol. PP, no. 99, pp. 1–1, 2015.

[11] F. Tobar, "Kernel-Based Adaptive Estimation: Multidimensional and State-Space Approaches," Ph.D. dissertation, Imperial College London, 2014.

[12] J. Mercer, "Functions of Positive and Negative Type, and their Connection with the Theory of Integral Equations," *Philosophical Trans. of the Royal Society of London (A)*, vol. 209, pp. 415–446, 1909.

[13] A. Shilton, "Mercer's theorem for quaternionic kernels," The University of Melbourn Department of Electrical and Electronic Engineering, Tech. Rep., 2007.

[14] J. Schürmann, *Pattern classification: A unified view of statistical and neural approaches*. Wiley, 1996.

[15] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.

[16] P. Bouboulis and S. Theodoridis, "Extension of Wirtinger's calculus to reproducing kernel Hilbert spaces and the complex kernel LMS," *IEEE Transactions on Signal Processing*, vol. 59, no. 3, pp. 964–978, Mar. 2011.