

Article

A Vector Representation of Multicomplex Numbers and Its Application to Radio Frequency Signals

Daniele Borio 

European Commission, Joint Research Centre, 21027 Ispra, VA, Italy; daniele.borio@ec.europa.eu

Abstract: Hypercomplex numbers, which are multi-dimensional extensions of complex numbers, have been proven beneficial in the development of advanced signal processing algorithms, including multi-dimensional filter design, linear regression and classification. We focus on multicomplex numbers, sets of hypercomplex numbers with commutative products, and introduce a vector representation allowing one to isolate the hyperbolic real and imaginary parts of a multicomplex number. The orthogonal decomposition of a multicomplex number is also discussed, and its connection with Hadamard matrices is highlighted. Finally, a multicomplex polar representation is provided. These properties are used to extend the standard complex baseband signal representation to the multi-dimensional case. It is shown that a set of 2^n Radio Frequency (RF) signals can be represented as the real part of a single multicomplex signal modulated by several frequencies. The signal RFs are related through a Hadamard matrix to the modulating frequencies adopted in the multicomplex baseband representation. Moreover, an orthogonal decomposition is provided for the obtained multicomplex baseband signal as a function of the complex baseband representations of the input RF signals.

Keywords: multicomplex numbers; Hadamard matrices; polar representation; orthogonal basis; radio frequency; baseband signals

MSC: 32-11; 94-XX; 94Axx; 94A12



Citation: Borio, D. A Vector Representation of Multicomplex Numbers and Its Application to Radio Frequency Signals. *Axioms* **2024**, *13*, 324. <https://doi.org/10.3390/axioms13050324>

Academic Editor: Dobrean Dan-Marius

Received: 11 April 2024

Revised: 9 May 2024

Accepted: 11 May 2024

Published: 14 May 2024



Copyright: © 2024 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Hypercomplex numbers are multi-dimensional extensions of complex numbers providing convenient representations of vector signals and images [1,2]. Hypercomplex numbers have been proven beneficial in the development of advanced signal processing algorithms, including multi-dimensional filter design [3,4], adaptive algorithm optimization [5–8], linear regression and classification [9]. While significant research effort has been devoted to quaternions [10,11], a four-dimensional extension of complex numbers, several families of hypercomplex numbers exist with different properties and are suitable for different applications [3,12,13]. Quaternions are effective in representing 3D rotations [10] and have the potential to capture relationships between different data channels [6]. However, they feature a non-commutative product, which is seldom reflected in signal processing applications. For this reason, commutative hypercomplex algebras have been considered [8,14,15]. In [15], a set of commutative reduced biquaternions is introduced: this is a four dimensional space with a commutative product. It is isomorphic to bicomplex numbers [16,17] and tessarines [18]. All these sets have the potential to provide compact and effective signal representations for four-dimensional signal processing. In the following, the term “bicomplex numbers” is used to denote four-dimensional commutative hypercomplex numbers. Bicomplex numbers belong to the family of multicomplex numbers [17,19], which are 2^{n-1} -dimensional extensions of complex numbers, where n defines the dimension of the specific multicomplex space. Complex numbers are obtained for $n = 1$, whereas bicomplex numbers are characterized by $n = 2$. Multicomplex numbers feature commutative products and are suitable for representing systems where the order of factors in a product should

not matter. For instance, bicomplex numbers are applied in Linear Time Invariant (LTI) system design [20] and dual-frequency Global Navigation Satellite System (GNSS) signal processing [21].

The purpose of this paper is twofold: first, to provide useful properties, including an effective vector representation, for manipulating multicomplex numbers, and second, to describe a possible application of multicomplex numbers for jointly representing signals from different RFs. These contributions enable advanced applications such as the derivation of multi-frequency acquisition and tracking algorithms for signal demodulation.

Multicomplex numbers are at first reviewed and a vector representation, in terms of pure imaginary and hyperbolic units [12,22], is provided. The orthogonal decomposition of a multicomplex number is also discussed and its connection with Hadamard matrices highlighted [12,23]. Finally, a multicomplex polar representation is provided. This representation involves vectors with imaginary and hyperbolic units and allows one to segregate hyperbolic modulus and phase components. These properties are used to extend the well-known baseband representation of an RF signal [24] to the multi-dimensional case. In one dimension, an RF signal can be represented as the real part of the product between a complex baseband signal and a complex exponential up-converting the signal to RF [24]. When two RF signals are considered, bicomplex numbers can be used to represent them as the real part of the product between a bicomplex signal and a bicomplex exponential [21]. This exponential is modulated by two frequencies: the first up-converts the bicomplex baseband signal to a common frequency, whereas the second splits it around the final RFs. Analogously, a set of 2^n RF signals can be represented as the real part of a single multicomplex signal modulated by several frequencies. These frequencies are related to the original signal RFs through the multiplication by a Hadamard matrix. Moreover, an orthogonal decomposition is provided for the multicomplex baseband signal as a function of the complex baseband representations of the input RF signals.

This work generalizes the analysis conducted in [21] that provided a baseband signal representation formula for the bicomplex case. In the bicomplex case, these representation formulas proved beneficial for algorithm design. The compact notation developed using bicomplex numbers eased algorithm development and result interpretation. The properties provided in the paper for multicomplex numbers have the potential to enable similar results in more general cases, for instance, with more signal components.

The remainder of this paper is organized as follows: Section 2 introduces multicomplex numbers, their representation in terms of hyperbolic and imaginary vectors and some basic properties. Section 3 discusses the orthogonal decomposition of a multicomplex number, whereas its polar representation is detailed in Section 4. Section 5 provides the multicomplex baseband representation of 2^n RF signals, and Section 6 concludes the paper.

2. Multicomplex Numbers

Multicomplex numbers are commutative algebras featuring the following:

- 1 real unit,
- 2^{n-1} imaginary units, which square to -1 ,
- $2^{n-1} - 1$ hyperbolic units, which square to 1.

Each unit multiplies a real coefficient, making \mathbb{C}_n isomorphic to \mathbb{R}^{2^n} . In the following, the notation introduced by [17] is adopted, and the symbol \mathbb{C}_n denotes a multicomplex space of order n . \mathbb{C}_1 is used to denote the algebra of complex numbers, whereas \mathbb{C}_2 denotes the space of bicomplex numbers [16].

\mathbb{C}_1 has one imaginary unit and no hyperbolic units, whereas bicomplex numbers are characterized by one real unit, two imaginary units and one hyperbolic unit.

In order to represent a multicomplex number, it is convenient to introduce two vectors containing the different real, imaginary and hyperbolic units generating \mathbb{C}_n . Let \mathbf{j}_n be the vector with 2^{n-1} imaginary units

$$\mathbf{j}_n = [j_0, j_1, \dots, j_{M-1}]^T, \quad \text{with } M = 2^{n-1}, \quad (1)$$

and \mathbf{k}_n be the vector with real and hyperbolic units

$$\mathbf{k}_n = [1, k_1, \dots, k_{M-1}]^T, \quad \text{with } M = 2^{n-1}. \tag{2}$$

Both \mathbf{j}_n and \mathbf{k}_n are column vectors of size $M \times 1$. A multicomplex number, $z_n \in \mathbb{C}_n$, can then be expressed as a function of \mathbf{j}_n and \mathbf{k}_n :

$$z_n = \mathbf{k}_n^T \mathbf{a}_M + \mathbf{j}_n^T \mathbf{b}_M, \tag{3}$$

where \mathbf{a}_M and \mathbf{b}_M are two real vectors of size $M \times 1$. For $n = 1$, standard complex numbers are found

$$z_1 = \mathbf{k}_1^T \mathbf{a}_1 + \mathbf{j}_1^T \mathbf{b}_1 = a + j_0 b, \tag{4}$$

where a and b are two real coefficients.

Vectors (1) and (2) can be constructed recursively. In particular,

- for $n = 1$,

$$\mathbf{j}_1 = [j_0]^T, \quad \mathbf{k}_1 = [1]^T \tag{5}$$

- for $n = 2$,

$$\begin{aligned} \mathbf{j}_2 &= [\mathbf{j}_1^T \quad j_1 \mathbf{k}_1^T]^T = [j_0 \quad j_1]^T \\ \mathbf{k}_2 &= [\mathbf{k}_1^T \quad j_1 \mathbf{j}_1^T]^T = [1 \quad j_1 j_0]^T = [1 \quad k_1]^T \end{aligned} \tag{6}$$

- for $n = 3$,

$$\begin{aligned} \mathbf{j}_3 &= [\mathbf{j}_2^T \quad j_2 \mathbf{k}_2^T]^T = [j_0 \quad j_1 \quad j_2 \quad j_2 k_1]^T = [j_0 \quad j_1 \quad j_2 \quad j_3]^T \\ \mathbf{k}_3 &= [\mathbf{k}_2^T \quad j_2 \mathbf{j}_2^T]^T = [1 \quad k_1 \quad j_2 j_0 \quad j_2 j_1]^T = [1 \quad k_1 \quad k_2 \quad k_3]^T \end{aligned} \tag{7}$$

- in general,

$$\begin{aligned} \mathbf{j}_n &= [\mathbf{j}_{n-1}^T \quad j_{2^{n-2}} \mathbf{k}_{n-1}^T]^T \\ \mathbf{k}_n &= [\mathbf{k}_{n-1}^T \quad j_{2^{n-2}} \mathbf{j}_{n-1}^T]^T. \end{aligned} \tag{8}$$

When passing from \mathbb{C}_{n-1} to \mathbb{C}_n , a new imaginary unit $j_{2^{n-2}}$, not available in \mathbb{C}_{n-1} , is introduced. The multiplication of $j_{2^{n-2}}$ with the elements of \mathbf{j}_{n-1} leads to 2^{n-2} new hyperbolic units, whereas the multiplication with the elements of \mathbf{k}_{n-1} leads to 2^{n-2} new imaginary units. This process is at the basis of the recursive construction of \mathbb{C}_n from \mathbb{C}_{n-1} (Chapter 5) of [17]. The validity of (8) can be proved by induction and requiring a commutative product for \mathbb{C}_n . The fact that the elements of $j_{2^{n-2}} \mathbf{k}_{n-1}$ are new imaginary units can be shown through simple algebraic manipulations. In particular, the square of the h th elements of $j_{2^{n-2}} \mathbf{k}_{n-1}$ is given by

$$(j_{2^{n-2}+h})^2 = (j_{2^{n-2}} k_h)^2 = j_{2^{n-2}} k_h j_{2^{n-2}} k_h = j_{2^{n-2}}^2 k_h^2 = -1, \tag{9}$$

where a commutative product has been assumed. By appending $j_{2^{n-2}} \mathbf{k}_{n-1}$ to \mathbf{j}_{n-1} , one finds the vector \mathbf{j}_n with all the imaginary units of \mathbb{C}_n . In (9), the index of $j_{2^{n-2}+h}$ is consistent with the ordering of the elements in \mathbf{j}_n . This proves the first part of (8). The vector with the real and hyperbolic units of \mathbb{C}_n is found in a similar way. Note that (3) provides a component-wise Cartesian representation of multicomplex numbers.

2.1. Unit Representation and Multiplication

From recursive construction (8), it emerges that the different imaginary and hyperbolic units are constructed as the product of basic imaginary units progressively introduced with the construction of higher-order multicomplex spaces. These units are

$$j_0, j_1, j_2, \dots, j_{2^{n-2}} \tag{10}$$

and \mathbb{C}_n is characterized by n such units. A notation where only basic units, (10), are used was adopted in [17]. In this paper, all the units are explicitly listed.

The recursive construction of the units of a multicomplex number (8) allows one to explicitly write a multicomplex unit as the product of the basic units. In particular,

$$k_i = j_0^{p_{n-1}(i)} \prod_{h=0}^{n-2} j_{2^h}^{b(i,h)}, \tag{11}$$

where $b(i, h)$ is the value of the $(h + 1)$ th digit of the binary representation of i

$$b(i, h) = \left\lfloor \frac{i}{2^h} \right\rfloor \bmod 2. \tag{12}$$

$p_{n-1}(i)$ is the parity bit associated to i . More specifically,

$$p_m(i) = \left(\sum_{h=0}^{m-1} b(i, h) \right) \bmod 2. \tag{13}$$

$p_{n-1}(i)$ is one if the number of 1s in the binary representation of i is odd and zero if it is even. For the binary representation, $n - 1$ bits are considered. As an example, consider \mathbb{C}_5 and $i = 13$. In \mathbb{C}_5 , there are 2^4 real/hyperbolic units, and i ranges from 0 to 15, with $k_0 = 1$. The binary representation of $i = 13$ is 1101; thus, $p_4(13) = 1$ and

$$k_{13} = j_8^1 j_4^1 j_2^0 j_1^1 j_0^1 = j_8 j_4 j_1 j_0.$$

Note that when elevated to 0, every number, including imaginary units, is equal to 1. A hyperbolic unit always requires the product of an even number of imaginary units: this is achieved through the parity function (13) and the introduction of j_0 , when required. Similarly to (11), an imaginary unit can be expressed as

$$j_i = j_0^{1-p_{n-1}(i)} \prod_{h=0}^{n-2} j_{2^h}^{b(i,h)}. \tag{14}$$

Equations (11) and (14) can be proven through induction. In particular, for $n = 2$

$$\begin{matrix} k_0 = 1 & j_0 = j_0 \\ k_1 = j_0 j_1 & j_1 = j_1 \end{matrix} \tag{15}$$

which trivially verifies (11) and (14). These conditions can be easily proven also for $n = 3$, when three basic units, j_0, j_1 and j_2 , are available:

$$\begin{matrix} k_0 = 1 & j_0 = j_0 \\ k_1 = j_0 j_1 & j_1 = j_1 \\ k_2 = j_2 j_0 & j_2 = j_2 \\ k_3 = j_2 j_1 & j_3 = j_0 j_1 j_2 \end{matrix} \tag{16}$$

Note that the first part of (16) coincides with (15). This is due to the recursive construction of multicomplex number spaces and to (8). The induction step can be verified assuming that (11) and (14) are true for $n - 1$ and using (8). As already noted for $n = 3$, the first part

of \mathbf{k}_n coincides with the elements of \mathbf{k}_{n-1} . These elements already verify (11) because of the inductive hypothesis. The second part of the vector is obtained by multiplying \mathbf{j}_{n-1} by $j_{2^{n-2}}$. The elements of this second part of the vector can be written as

$$k_{2^{n-2}+i} = j_{2^{n-2}}j_i = j_{2^{n-2}}j_0^{1-p_{n-2}(i)} \prod_{h=0}^{n-3} j_{2^h}^{b(i,h)}. \tag{17}$$

The index $2^{n-2} + i$ has an extra bit set to 1 with respect to i , and its parity is equal to $1 - p_{n-2}(i)$. Moreover, the multiplication by $j_{2^{n-2}}$ can be included in the main product in (17). Thus,

$$k_{2^{n-2}+i} = j_0^{p_{n-1}(2^{n-2}+i)} \prod_{h=0}^{n-2} j_{2^h}^{b(2^{n-2}+i,h)}. \tag{18}$$

This proves the induction step also for the elements in the second part of \mathbf{k}_n . The induction step for the elements of \mathbf{j}_n is proven in a similar way.

From (11) and (14), it follows that the ratio between j_i and k_i is equal to

$$\frac{j_i}{k_i} = \frac{j_0^{1-p_{n-1}(i)}}{j_0^{p_{n-1}(i)}} = j_0 j_0^{-2p_{n-1}(i)} = (-1)^{p_{n-1}(i)} j_0. \tag{19}$$

Equations (11) and (14) also provide a simple rule for multiplying the different units. More specifically,

$$\begin{aligned} k_i k_l &= j_0^{p_{n-1}(i)+p_{n-1}(l)} \prod_{h=0}^{n-2} j_{2^h}^{b(i,h)+b(l,h)} \\ &= (-1)^{s(i,l)} k_{i \oplus l}, \end{aligned} \tag{20}$$

where $i \oplus l$ is the integer resulting from the bitwise XORing of the bits of i and l . $s(i, l)$ is the number of common ones in the binary representations of i and l including the parity bits:

$$s(i, j) = \sum_{h=0}^{n-2} b(i, h)b(l, h) + p_{n-1}(i)p_{n-1}(l). \tag{21}$$

A similar multiplication rule is obtained for the imaginary units

$$\begin{aligned} j_i j_l &= j_0^{2-p_{n-1}(i)-p_{n-1}(l)} \prod_{h=0}^{n-2} j_{2^h}^{b(i,h)+b(l,h)} \\ &= -(-1)^{p_{n-1}(i)+p_{n-1}(l)+s(i,l)} k_{i \oplus l} \end{aligned} \tag{22}$$

and for the cross-product:

$$\begin{aligned} j_i k_l &= (-1)^{p_{n-1}(i)} j_0 k_i k_l = (-1)^{p_{n-1}(i)} j_0 (-1)^{s(i,l)} k_{i \oplus l} \\ &= -(-1)^{p_{n-1}(i)+s(i,l)} j_{i \oplus l}. \end{aligned} \tag{23}$$

2.2. Multiplication by j_0

An important property is the multiplication of vectors (1) and (2) with j_0 , the first basic imaginary unit defining complex numbers. In particular,

$$\begin{aligned} \mathbf{j}_n &= j_0 \mathbf{D}_n \mathbf{k}_n, & j_0 \mathbf{j}_n &= -\mathbf{D}_n \mathbf{k}_n \\ \mathbf{k}_n &= -j_0 \mathbf{D}_n \mathbf{j}_n, & j_0 \mathbf{k}_n &= \mathbf{D}_n \mathbf{j}_n, \end{aligned} \tag{24}$$

where \mathbf{D}_n is a diagonal matrix built recursively as

$$\mathbf{D}_n = \begin{bmatrix} \mathbf{D}_{n-1} & \mathbf{0}_{2^{n-2} \times 2^{n-2}} \\ \mathbf{0}_{2^{n-2} \times 2^{n-2}} & -\mathbf{D}_{n-1} \end{bmatrix} \tag{25}$$

and $\mathbf{0}_{a \times b}$ denotes a zero matrix of size $a \times b$. For $n = 1$, $\mathbf{D}_1 = 1$. The proof of (24) is done by induction. For $n = 1$,

$$\mathbf{j}_1 = [j_0] \quad \mathbf{k}_1 = [1]$$

and trivially

$$j_0 \mathbf{j}_1 = j_0 \cdot j_0 = -1 = -\mathbf{D}_1 \mathbf{k}_1,$$

which is the base case of the first equation in (24). Note that the two conditions in the two equations in (24) are equivalent and one follows directly from the other through multiplication by j_0 and recalling that $j_0^2 = -1$. Similarly,

$$j_0 \mathbf{k}_1 = j_0 \cdot 1 = j_0 = \mathbf{D}_1 \mathbf{j}_1,$$

which is the proof of the base case for the second condition in (24).

To prove the induction step, one assumes that (24) is true for $n - 1$ and uses recursive formulas (8). In particular,

$$\begin{aligned} j_0 \mathbf{j}_n &= j_0 [\mathbf{j}_{n-1}^T \quad j_{2^{n-2}} \mathbf{k}_{n-1}^T]^T = [j_0 \mathbf{j}_{n-1}^T \quad j_0 j_{2^{n-2}} \mathbf{k}_{n-1}^T]^T \\ &= [-(\mathbf{D}_{n-1} \mathbf{k}_{n-1})^T \quad j_{2^{n-2}} (\mathbf{D}_{n-1} \mathbf{j}_{n-1})^T]^T \\ &= - \begin{bmatrix} \mathbf{D}_{n-1} \mathbf{k}_{n-1} \\ -\mathbf{D}_{n-1} j_{2^{n-2}} \mathbf{j}_{n-1} \end{bmatrix} \\ &= - \begin{bmatrix} \mathbf{D}_{n-1} & \mathbf{0}_{2^{n-2} \times 2^{n-2}} \\ \mathbf{0}_{2^{n-2} \times 2^{n-2}} & -\mathbf{D}_{n-1} \end{bmatrix} \begin{bmatrix} \mathbf{k}_{n-1} \\ j_{2^{n-2}} \mathbf{j}_{n-1} \end{bmatrix} \\ &= -\mathbf{D}_n \mathbf{k}_n. \end{aligned} \tag{26}$$

Equation (26) completes the proof for the first part of (24). The induction step for the second condition is found in a similar way. The definition of \mathbf{D}_n in (25) follows directly from the recursive construction of the imaginary and hyperbolic vectors.

Multiplication formula (24) allows one to express \mathbf{k}_n as a function of \mathbf{k}_{n-1} only. Starting from the second equation in (8), one has:

$$\mathbf{k}_n = \begin{bmatrix} \mathbf{k}_{n-1} \\ j_{2^{n-2}} \mathbf{j}_{n-1} \end{bmatrix} = \begin{bmatrix} \mathbf{k}_{n-1} \\ j_{2^{n-2}} j_0 \mathbf{D}_{n-1} \mathbf{k}_{n-1} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{2^{n-2}} \\ k_{2^{n-2}} \mathbf{D}_{n-1} \end{bmatrix} \mathbf{k}_{n-1}, \tag{27}$$

where the first equation in (24) has been used to express \mathbf{j}_{n-1} in terms of \mathbf{k}_{n-1} . Also note that from the multiplication rules introduced in the previous section, $k_{2^{n-2}} = j_{2^{n-2}} j_0$. In (27), \mathbf{I}_a denotes the identity matrix of size $a \times a$.

The hyperbolic units in \mathbf{k}_n define hyperbolic spaces of dimension 2^{n-1} [22]. Hyperbolic spaces are closed with respect to addition and multiplication and can be constructed recursively using (27). In the following, the notation \mathbb{D}_n will be used to denote a hyperbolic space characterized by 1 real unit and $2^{n-1} - 1$ hyperbolic units. For instance, \mathbb{D}_2 denotes the set of hyperbolic numbers [22]. \mathbb{D}_1 coincides with the set of real numbers.

2.3. Conjugation, Hyperbolic Real and Imaginary Parts

Since there are several imaginary and hyperbolic units, it is possible to define several conjugation operations for multicomplex numbers [8,16]. In the following, the *-conjugation [16], which extends the standard conjugation for complex numbers, is considered. It is defined as follows:

$$z_n^* = \mathbf{k}_n^T \mathbf{a}_M - \mathbf{j}_n^T \mathbf{b}_M \tag{28}$$

and it is obtained by negating the sign of all imaginary units. Using (28), one can obtain the multicomplex equivalent of the real and imaginary parts of a complex number. In particular,

$$\Re_k\{z_n\} = \frac{1}{2}[z_n + z_n^*] = \mathbf{k}_n^T \mathbf{a}_M \tag{29}$$

and

$$\Im_k\{z_n\} = \frac{1}{2j_0}[z_n - z_n^*] = \frac{1}{j_0} \mathbf{j}_n^T \mathbf{b}_M = \mathbf{k}_n^T \mathbf{D}_n \mathbf{b}_M. \tag{30}$$

In (29) (and for analogy in (30)), an index ‘k’ has been added to distinguish the real part operator, $\Re\{\cdot\}$, which isolates in a multicomplex number the real coefficient multiplying the real unit, 1, from $\mathbf{k}_n^T \mathbf{a}_M$. In order to avoid confusion, (29) and (30) are denoted as hyperbolic real and hyperbolic imaginary parts of a multicomplex number. Note that both $\Re_k\{z_n\}$ and $\Im_k\{z_n\}$ belong to \mathbb{D}_n , the 2^{n-1} -dimensional hyperbolic space defined by \mathbf{k}_n [22]. As already introduced above, \mathbb{D}_n is a subspace of \mathbb{C}_n and is closed with respect to addition and multiplication. With respect to multicomplex numbers, \mathbb{D}_n plays a role similar to that of real numbers for complex numbers.

2.4. Multiplication of Multicomplex Numbers

Relationship (24) allows one to express a multicomplex number in terms of \mathbf{k}_n and j_0 only. In particular,

$$z_n = \mathbf{k}_n^T \mathbf{a}_M + j_0 \mathbf{k}_n^T \mathbf{D}_n \mathbf{b}_M = \mathbf{k}_n^T [\mathbf{a}_M + j_0 \mathbf{D}_n \mathbf{b}_M] = \mathbf{k}_n^T \mathbf{z}_M, \tag{31}$$

which is the scalar product between the complex vector, \mathbf{z}_M , and \mathbf{k}_n . Equation (31) can be used to perform multiplication between multicomplex numbers. Given two multicomplex numbers, z_n and y_n , then

$$z_n \cdot y_n = \mathbf{k}_n^T \mathbf{z}_M \mathbf{k}_n^T \mathbf{y}_M = \mathbf{z}_M^T \mathbf{k}_n \mathbf{k}_n^T \mathbf{y}_M = \mathbf{z}_M^T \mathbf{K}_n \mathbf{y}_M, \tag{32}$$

where $\mathbf{K}_n = \mathbf{k}_n \mathbf{k}_n^T$ is a square symmetric matrix whose diagonal elements are equal to 1 and whose rows and columns are permutations of \mathbf{k}_n with possible sign changes. \mathbf{K}_n is essentially the multiplication table of the hyperbolic units. For instance,

$$\mathbf{K}_2 = \begin{bmatrix} 1 & k_1 \\ k_1 & 1 \end{bmatrix} \tag{33}$$

and

$$\mathbf{K}_3 = \begin{bmatrix} 1 & k_1 & k_2 & k_3 \\ k_1 & 1 & -k_3 & -k_2 \\ k_2 & -k_3 & 1 & -k_1 \\ k_3 & -k_2 & -k_1 & 1 \end{bmatrix}. \tag{34}$$

\mathbf{K}_n can be constructed either using (20) or recursively as

$$\mathbf{K}_n = \begin{bmatrix} \mathbf{K}_{n-1} & k_{2^{n-2}} \mathbf{K}_{n-1} \mathbf{D}_{n-1} \\ k_{2^{n-2}} \mathbf{D}_{n-1} \mathbf{K}_{n-1} & \mathbf{D}_{n-1} \mathbf{K}_{n-1} \mathbf{D}_{n-1} \end{bmatrix}. \tag{35}$$

\mathbf{K}_n is directly related to the matrix representation of a multicomplex number and can be used to implement multicomplex multiplication in software [25]. Let us denote w_n as the result of the product between two multicomplex numbers, z_n and y_n ; then, $w_n = \mathbf{k}_n^T \mathbf{w}_M$ with \mathbf{w}_M the vector containing the complex components of w_n . It is possible to show that

$$\mathbf{w}_M = \mathbf{Z}_n \mathbf{y}_M \tag{36}$$

where \mathbf{Z}_n is a $2^{n-1} \times 2^{n-1}$ symmetric matrix obtained by replacing the elements of \mathbf{K}_n with the corresponding coefficients in \mathbf{z}_M : 1s are replaced by the first element in \mathbf{z}_M , k_1 with the second, k_2 with the third and so on. \mathbf{Z}_n is a matrix representation of z_n .

3. Orthogonal Decomposition

Multicomplex numbers are characterized by sets of orthogonal idempotent units, which allow the decomposition of a multicomplex number into orthogonal components [17]. The set of bicomplex numbers \mathbb{C}_2 is characterized by 2 orthogonal idempotent units [16]

$$e_{2,0} = \frac{1 - k_1}{2}, \quad e_{2,1} = \frac{1 + k_1}{2}. \tag{37}$$

Any power of $e_{2,0}$ and $e_{2,1}$ is equal to the original base value, either $e_{2,0}$ or $e_{2,1}$. Moreover, $e_{2,0}e_{2,1} = 0$. Any bicomplex number can be expressed as

$$z_2 = z_{2,0}e_{2,0} + z_{2,1}e_{2,1}, \tag{38}$$

where $z_{2,0}$ and $z_{2,1} \in \mathbb{C}_1$. Orthogonal decomposition (38) allows one to perform operations on bicomplex numbers in terms of the orthogonal components $z_{2,0}$ and $z_{2,1}$.

For higher-dimensional spaces, the set of orthogonal idempotent units can be constructed recursively [17]. Moreover, since there exists several hyperbolic units, it is possible to construct several sets of orthogonal idempotent units. In the following, we use the set containing the following units:

$$e_{n,i} = \prod_{h=0}^{n-2} \frac{1 - (-1)^{b(i,h)}k_{2^h}}{2} \quad i = 0, \dots, 2^{n-2}, \tag{39}$$

where $b(i, h)$ has been defined in (12). For instance, for $n = 3$, four orthogonal idempotent units are found:

$$\begin{aligned} e_{3,0} &= \frac{1 - k_2}{2} \frac{1 - k_1}{2}, & e_{3,1} &= \frac{1 - k_2}{2} \frac{1 + k_1}{2} \\ e_{3,2} &= \frac{1 + k_2}{2} \frac{1 - k_1}{2}, & e_{3,3} &= \frac{1 + k_2}{2} \frac{1 + k_1}{2}. \end{aligned} \tag{40}$$

Note that (39) can be implemented in a recursive way:

$$\begin{aligned} e_{n,i} &= e_{n-1,i} \frac{1 - k_{2^{n-2}}}{2} \\ e_{n,i+2^{n-2}} &= e_{n-1,i} \frac{1 + k_{2^{n-2}}}{2}. \end{aligned} \tag{41}$$

The 2^{n-1} orthogonal idempotent units can be arranged in a vector

$$\mathbf{e}_n = [e_{n,0}, e_{n,1}, \dots, e_{n,2^{n-1}-1}]^T, \tag{42}$$

which has the same size of \mathbf{k}_n . \mathbf{e}_n can be reconstructed recursively as

$$\mathbf{e}_n = \begin{bmatrix} \mathbf{e}_{n-1} \frac{1 - k_{2^{n-2}}}{2} \\ \mathbf{e}_{n-1} \frac{1 + k_{2^{n-2}}}{2} \end{bmatrix} = \mathbf{e}_{n-1} \otimes \begin{bmatrix} \frac{1 - k_{2^{n-2}}}{2} \\ \frac{1 + k_{2^{n-2}}}{2} \end{bmatrix} \tag{43}$$

where \otimes denotes the Kronecker product. The same equation can be written using the standard matrix product as follows:

$$\mathbf{e}_n = \begin{bmatrix} \mathbf{I}_{2^{n-2}} \frac{1 - k_{2^{n-2}}}{2} \\ \mathbf{I}_{2^{n-2}} \frac{1 + k_{2^{n-2}}}{2} \end{bmatrix} \mathbf{e}_{n-1}. \tag{44}$$

For $n = 2$, it is simple to verify that

$$\mathbf{k}_2 = \begin{bmatrix} 1 \\ k_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \mathbf{e}_2 = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \frac{1-k_1}{2} \\ \frac{1+k_1}{2} \end{bmatrix}. \tag{45}$$

The last matrix appearing in (45) can be expressed as

$$\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \mathbf{D}_2 \mathbf{H}_2, \tag{46}$$

where \mathbf{D}_2 is the diagonal matrix introduced in (25) and \mathbf{H}_2 is a Hadamard matrix of order 2 [23]. A Hadamard matrix is a square matrix whose entries are equal to either 1 or -1 . Moreover, its rows are mutually orthogonal. There exists several construction procedures for obtaining Hadamard matrices that are considered equivalent if they lead to matrices differing only for sign changes in the rows or columns or for row or column permutations. The construction rule adopted here is specified in the following. Thus, (45) can be expressed as

$$\mathbf{k}_2 = \mathbf{D}_2 \mathbf{H}_2 \mathbf{e}_2 \tag{47}$$

or equivalently

$$\mathbf{e}_2 = \frac{1}{2} (\mathbf{D}_2 \mathbf{H}_2)^T \mathbf{k}_2, \tag{48}$$

where the following properties have been exploited

$$\mathbf{D}_n^{-1} = \mathbf{D}_n = \mathbf{D}_n^T \tag{49}$$

and

$$\mathbf{H}_M^{-1} = \frac{1}{M} \mathbf{H}_M^T. \tag{50}$$

A relationship similar to (47) exists for $n = 3$. In particular,

$$\mathbf{k}_3 = \begin{bmatrix} 1 \\ k_1 \\ k_2 \\ k_3 \end{bmatrix} \quad \text{and} \quad \mathbf{e}_3 = \begin{bmatrix} \frac{1-k_1}{2} & \frac{1-k_2}{2} \\ \frac{1+k_1}{2} & \frac{1-k_2}{2} \\ \frac{1-k_1}{2} & \frac{1+k_2}{2} \\ \frac{1+k_1}{2} & \frac{1+k_2}{2} \end{bmatrix}. \tag{51}$$

These two vectors are related by

$$\mathbf{k}_3 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & -1 \end{bmatrix} \mathbf{e}_3 \tag{52}$$

where

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ -1 & 1 & 1 & -1 \end{bmatrix} = \mathbf{D}_3 \mathbf{H}_4. \tag{53}$$

\mathbf{H}_4 is a Hadamard matrix of order $4 = 2^{3-1}$. Thus,

$$\mathbf{k}_3 = \mathbf{D}_3 \mathbf{H}_4 \mathbf{e}_3 \quad \text{or} \quad \mathbf{e}_3 = \frac{1}{4} (\mathbf{D}_3 \mathbf{H}_4)^T \mathbf{k}_3 \tag{54}$$

Equations (48) and (54) suggest that there is a simple relationship between the vector of the hyperbolic units, \mathbf{k}_n , and the one of the orthogonal basis \mathbf{e}_n . The general relationship

$$\mathbf{e}_n = \frac{1}{2^{n-1}} (\mathbf{D}_n \mathbf{H}_{2^{n-1}})^T \mathbf{k}_n \tag{55}$$

can be proven by induction if the following construction rule is adopted for $\mathbf{H}_{2^{n-1}}$:

$$\mathbf{H}_{2^{n-1}} = \begin{bmatrix} \mathbf{H}_{2^{n-2}} & \mathbf{H}_{2^{n-2}} \\ \mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}} & -\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}} \end{bmatrix}. \tag{56}$$

The only difference with respect to the Sylvester’s construction for Hadamard matrices [23] is the presence of the diagonal matrix \mathbf{D}_{n-1} , which contains only 1s and -1 s and introduces some sign changes for the rows of the resulting matrices.

The base step has already been proven for $n = 2$. The induction step is proven starting from (44):

$$\begin{aligned} \mathbf{e}_n &= \begin{bmatrix} \mathbf{I}_{2^{n-2}} \frac{1-k_{2^{n-2}}}{2} \\ \mathbf{I}_{2^{n-2}} \frac{1+k_{2^{n-2}}}{2} \end{bmatrix} \mathbf{e}_{n-1} = \frac{1}{2} \begin{bmatrix} \mathbf{I}_{2^{n-2}} & -\mathbf{I}_{2^{n-2}} \\ \mathbf{I}_{2^{n-2}} & \mathbf{I}_{2^{n-2}} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{2^{n-2}} \\ \mathbf{I}_{2^{n-2}} k_{2^{n-2}} \end{bmatrix} \mathbf{e}_{n-1} \\ &= \frac{1}{2} \begin{bmatrix} \mathbf{I}_{2^{n-2}} & -\mathbf{I}_{2^{n-2}} \\ \mathbf{I}_{2^{n-2}} & \mathbf{I}_{2^{n-2}} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{2^{n-2}} \\ \mathbf{I}_{2^{n-2}} k_{2^{n-2}} \end{bmatrix} \left(\frac{1}{2^{n-2}} \mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}} \right)^T \mathbf{k}_{n-1}, \end{aligned} \tag{57}$$

where (55) has been assumed to be true for $n - 1$. By performing the different matrix products, (57) becomes

$$\begin{aligned} \mathbf{e}_n &= \frac{1}{2^{n-1}} \begin{bmatrix} \mathbf{I}_{2^{n-2}} & -\mathbf{I}_{2^{n-2}} \\ \mathbf{I}_{2^{n-2}} & \mathbf{I}_{2^{n-2}} \end{bmatrix} \begin{bmatrix} (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T \mathbf{k}_{n-1} \\ (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T k_{2^{n-2}} \mathbf{k}_{n-1} \end{bmatrix} \\ &= \frac{1}{2^{n-1}} \begin{bmatrix} (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T \mathbf{k}_{n-1} - (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T k_{2^{n-2}} \mathbf{k}_{n-1} \\ (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T \mathbf{k}_{n-1} + (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T k_{2^{n-2}} \mathbf{k}_{n-1} \end{bmatrix} \\ &= \frac{1}{2^{n-1}} \begin{bmatrix} (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T [\mathbf{I}_{2^{n-2}} - \mathbf{D}_{n-1} k_{2^{n-2}} \mathbf{D}_{n-1}] \mathbf{k}_{n-1} \\ (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T [\mathbf{I}_{2^{n-2}} + \mathbf{D}_{n-1} k_{2^{n-2}} \mathbf{D}_{n-1}] \mathbf{k}_{n-1} \end{bmatrix} \\ &= \frac{1}{2^{n-1}} \begin{bmatrix} (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T & -(\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T \mathbf{D}_{n-1} \\ (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T & (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T \mathbf{D}_{n-1} \end{bmatrix} \begin{bmatrix} \mathbf{I}_{2^{n-1}} \\ k_{2^{n-2}} \mathbf{D}_{n-1} \end{bmatrix} \mathbf{k}_{n-1} \\ &= \frac{1}{2^{n-1}} \begin{bmatrix} (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T & -\mathbf{H}_{2^{n-2}}^T \\ (\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}})^T & \mathbf{H}_{2^{n-2}}^T \end{bmatrix} \mathbf{k}_n, \end{aligned} \tag{58}$$

where (27) and the properties of \mathbf{D}_n have been exploited.

It is easy to verify that the matrix in the last line of (58) is the transpose of $\mathbf{D}_n \mathbf{H}_{2^{n-1}}$:

$$\begin{aligned} \mathbf{D}_n \mathbf{H}_{2^{n-1}} &= \begin{bmatrix} \mathbf{D}_{n-1} & \mathbf{0}_{2^{n-2} \times 2^{n-2}} \\ \mathbf{0}_{2^{n-2} \times 2^{n-2}} & -\mathbf{D}_{n-1} \end{bmatrix} \begin{bmatrix} \mathbf{H}_{2^{n-2}} & \mathbf{H}_{2^{n-2}} \\ \mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}} & -\mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}} & \mathbf{D}_{n-1} \mathbf{H}_{2^{n-2}} \\ -\mathbf{H}_{2^{n-2}} & \mathbf{H}_{2^{n-2}} \end{bmatrix}. \end{aligned} \tag{59}$$

This completes the proof of (55) and of the inverse relationship

$$\mathbf{k}_n = \mathbf{D}_n \mathbf{H}_{2^{n-1}} \mathbf{e}_n. \tag{60}$$

Using (60), it is possible to express z_n in terms of its orthogonal components. More specifically,

$$z_n = \mathbf{e}_n^T (\mathbf{D}_n \mathbf{H}_{2^{n-1}})^T \mathbf{z}_M = \mathbf{e}_n^T \mathbf{z}_M^O, \tag{61}$$

where

$$\mathbf{z}_M^O = (\mathbf{D}_n \mathbf{H}_{2^{n-1}})^T \mathbf{z}_M \tag{62}$$

is the vector with the orthogonal components of z_n . \mathbf{z}_M^O is obtained through a Hadamard transform of the original vector \mathbf{z}_M , and its elements belong to \mathbb{C}_1 .

Equation (61) provides an alternative way to perform multiplication between multi-complex numbers:

$$w_n = z_n \cdot y_n = \left(\mathbf{z}_M^O\right)^T \mathbf{e}_n \mathbf{e}_n^T \mathbf{y}_M^O = \mathbf{e}_n^T \mathbf{w}_M^O, \tag{63}$$

where the components of the vector \mathbf{w}_M^O are obtained as the product of the corresponding elements in \mathbf{z}^O and \mathbf{y}^O . Note that $\mathbf{E}_n = \mathbf{e}_n \mathbf{e}_n^T$ is a square diagonal matrix whose elements are the idempotent units of \mathbf{e}_n . This is why it is possible to operate component-wise in the product (63). Orthogonal component decomposition can also be exploited to compute several functions of multicomplex numbers defined, for instance, in terms of convergent power series [26].

4. Polar Representation

In the previous section, a Cartesian representation, in terms of hyperbolic real and imaginary parts, was given for multicomplex numbers. Similarly to complex numbers, it is also possible to provide a polar representation for z_n . In particular, it is possible to express a multicomplex number as

$$z_n = \mathbf{k}_n^T \mathbf{r}_M \exp\left\{j_n^T \Phi_M\right\} = \mathbf{k}_n^T \mathbf{r}_M \exp\left\{j_0 \mathbf{k}_n^T \mathbf{D}_n \Phi_M\right\}, \tag{64}$$

where \mathbf{r}_M and Φ_M are real vectors of size $M = 2^{n-1}$ containing modulus and phase information, respectively. Product $\mathbf{k}_n^T \mathbf{r}_M$ defines the hyperbolic modulus $\in \mathbb{D}_n$, whereas $\mathbf{k}_n^T \mathbf{D}_n \Phi_M$ is the hyperbolic phase also belonging to \mathbb{D}_n . Note that exponential and trigonometric functions maintain their usual properties when evaluated on multicomplex numbers [26]. A proof of these properties, which include Euler’s formulas, can be found in [26].

Euler’s formula can be used to compute the hyperbolic real and imaginary parts from the hyperbolic modulus and phase. More specifically, it follows that

$$\begin{aligned} \mathbf{k}_n^T \mathbf{a}_M &= \mathbf{k}_n^T \mathbf{r}_M \cos\left(\mathbf{k}_n^T \mathbf{D}_n \Phi_M\right) \\ \mathbf{k}_n^T \mathbf{D}_n \mathbf{b}_M &= \mathbf{k}_n^T \mathbf{r}_M \sin\left(\mathbf{k}_n^T \mathbf{D}_n \Phi_M\right) \end{aligned} \tag{65}$$

The terms on the right-hand sides of (65) are products of multi-dimensional hyperbolic numbers. The cosine and sine functions can be computed either recursively, using trigonometric addition and subtraction formulas [25] or using the orthogonal decomposition of a multicomplex number.

The hyperbolic modulus $\mathbf{k}_n^T \mathbf{r}_M$ can be obtained through conjugation (28). Note that

$$z_n^* = \mathbf{k}_n^T \mathbf{r}_M \exp\left\{-j_0 \mathbf{k}_n^T \mathbf{D}_n \Phi_M\right\}. \tag{66}$$

Thus,

$$|z_n|_k^2 = z_n \cdot z_n^* = (\mathbf{k}_n^T \mathbf{r}_M)^2 = \mathbf{z}_M^T \mathbf{K}_n \mathbf{z}_M^* = \mathbf{k}_n^T \mathbf{Z}_n \mathbf{z}_M^*, \tag{67}$$

where conjugation is applied to the components of \mathbf{z}_M and \mathbf{Z}_n is the matrix obtained from \mathbf{z}_M according to the procedure detailed in Section 2.4. A more efficient approach for determining the components of $\mathbf{k}_n^T \mathbf{r}_M$ is to use the orthogonal decomposition of a multicomplex number. From (63), it follows that the orthogonal components of $|z_n|_k^2$ are the square moduli of the elements of \mathbf{z}_M^O . Similarly, the orthogonal components of $\mathbf{k}_n^T \mathbf{r}_M$ are the moduli of the orthogonal components of \mathbf{z}_M .

In a similar way, it is possible to show that the hyperbolic phase vector Φ_M can be obtained from the phases of the orthogonal components of \mathbf{z}_M . Let

$$\Phi_M^O = \begin{bmatrix} \angle z_{M,0}^O \\ \angle z_{M,1}^O \\ \vdots \\ \angle z_{M,M-1}^O \end{bmatrix} \tag{68}$$

be the vector with the phases of the orthogonal components of \mathbf{z}_M that are the phases of the elements of \mathbf{z}_M^O . The hyperbolic phase vector is finally computed as

$$\Phi_M = \frac{1}{2^{n-1}} \mathbf{H}_{2^{n-1}} \Phi_M^O \tag{69}$$

Equation (69) is obtained from (62) by taking into account that Φ_M is referred to \mathbf{j}_n instead of \mathbf{k}_n .

5. Baseband Representation of 2^{n-1} RF Signals

A single RF signal modulated at f_0

$$x_{RF}(t) = x_I(t) \cos(2\pi f_0 t) + x_Q(t) \sin(2\pi f_0 t) \tag{70}$$

can always be expressed as [24]

$$x_{RF}(t) = \Re\{x(t) \exp\{j_0 2\pi f_0 t\}\} \tag{71}$$

with $x(t) = x_I(t) - j_0 x_Q(t) \in \mathbb{C}_1$. $x(t)$ is the complex baseband representation of $x_{RF}(t)$, and $\Re\{\cdot\}$ is the real part operator. As already discussed, $\Re\{\cdot\}$ isolates the coefficient multiplying the real unit of a (multi)complex number and should not be confused with the hyperbolic real part $\Re_k\{\cdot\}$. Similarly, when two RF signals, $x_{0,RF}(t)$ and $x_{1,RF}(t)$, are considered, it is possible to obtain the following bicomplex baseband representation [21]:

$$\begin{aligned} z_{2,RF}(t) &= x_{0,RF}(t) + x_{1,RF}(t) \\ &= \Re\{x_0(t) \exp\{j_0 2\pi f_{X,0} t\}\} + \Re\{x_1(t) \exp\{j_0 2\pi f_{X,1} t\}\} \\ &= \Re\{[x_0(t) \exp\{j_0 2\pi f_1 t\} + x_1(t) \exp\{-j_0 2\pi f_1 t\}] \exp\{j_0 2\pi f_0 t\}\}, \end{aligned} \tag{72}$$

where $f_{X,0}$ and $f_{X,1}$ are two RFs and $x_0(t)$ and $x_1(t)$ the complex baseband versions of $x_{0,RF}(t)$ and $x_{1,RF}(t)$, respectively. The following transformation has been introduced on the two RFs $f_{X,0}$ and $f_{X,1}$:

$$\begin{bmatrix} f_0 \\ f_1 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} f_{X,0} \\ f_{X,1} \end{bmatrix} = \frac{1}{2} \mathbf{H}_2 \begin{bmatrix} f_{X,0} \\ f_{X,1} \end{bmatrix} \tag{73}$$

By expanding the terms in square brackets in (72), one finds

$$z_{2,RF}(t) = \Re\{[[x_0(t) + x_1(t)] \cos(2\pi f_1 t) + j_0[x_0(t) - x_1(t)] \sin(2\pi f_1 t)] \exp\{j_0 2\pi f_0 t\}\} \tag{74}$$

The term in the outer square brackets in (74) can be written as the real part of the product of two bicomplex numbers. In particular, let us define

$$\begin{aligned} z_{2,I}(t) &= x_0(t) + x_1(t) \\ z_{2,Q}(t) &= j_0[x_0(t) - x_1(t)] \\ z_2(t) &= z_{2,I}(t) - j_1 z_{2,Q}(t) \in \mathbb{C}_2, \end{aligned} \tag{75}$$

then

$$z_{2,RF}(t) = \Re\{z(t) \exp\{j12\pi f_1 t\} \exp\{j02\pi f_0 t\}\} = \Re\left\{z(t) \exp\left\{2\pi j \frac{1}{2} \mathbf{f}_2 t\right\}\right\} \tag{76}$$

where $\mathbf{f}_2 = [f_0 \ f_1]^T$.

We want to prove that, for an arbitrary n , the sum of 2^{n-1} RF signals can be expressed as

$$z_{n,RF} = \Re\left\{z_n(t) \exp\left\{2\pi j \mathbf{f}_n^T \mathbf{f}_n t\right\}\right\}, \tag{77}$$

where $z_{n,RF}(t) = \sum_{i=0}^{2^{n-1}-1} x_{i,RF}(t)$ and $z_n(t) \in \mathbb{C}_n$. \mathbf{f}_n is a vector of 2^{n-1} frequencies. Each real signal, $x_{i,RF}(t)$, is modulated into a different RF, $f_{X,i}$, and \mathbf{f}_n is obtained through a linear transformation of the vector of the original RFs:

$$\mathbf{f}_n = \frac{1}{2^{n-1}} \mathbf{H}_{2^{n-1}} \mathbf{f}_{X,n} = \frac{1}{2^{n-1}} \mathbf{H}_{2^{n-1}} \begin{bmatrix} f_{X,0} \\ f_{X,1} \\ \vdots \\ f_{X,2^{n-1}-1} \end{bmatrix}, \tag{78}$$

where $\mathbf{H}_{2^{n-1}}$ is the Hadamard matrix defined in (56). For $n = 1$, $\mathbf{H}_1 = 1$, whereas \mathbf{H}_2 is found in the right side of (73). A proof of (77) is given by induction: the base step has already been proved for $n = 1$ and $n = 2$. The induction step assumes as true (77) is true for $n = k$. Under this assumption, it is necessary to prove (77) for $n = k + 1$. To do so, it is possible to split $z_{k+1,RF}(t)$ as the sum of two groups of RF signals:

$$z_{k+1,RF}(t) = \sum_{i=0}^{2^{k-1}-1} x_{i,RF}(t) + \sum_{i=0}^{2^{k-1}-1} x_{i+2^{k-1},RF}(t). \tag{79}$$

The first summation, $\sum_{i=0}^{2^{k-1}-1} x_{i,RF}(t)$, is denoted $z_{k,RF}^0(t)$, whereas the second term is denoted as $z_{k,RF}^1(t)$. Thus,

$$z_{k+1,RF}(t) = z_{k,RF}^0(t) + z_{k,RF}^1(t). \tag{80}$$

The inductive hypothesis can be applied to both components in (80):

$$z_{k+1,RF}(t) = \Re\left\{z_k^0(t) e^{2\pi j \mathbf{f}_{k,0}^T t}\right\} + \Re\left\{z_k^1(t) e^{2\pi j \mathbf{f}_{k,1}^T t}\right\}, \tag{81}$$

where both $z_k^0(t)$ and $z_k^1(t) \in \mathbb{C}_k$. The frequency vectors in (81) are obtained as

$$\mathbf{f}_{k,0} = \frac{1}{2^{k-1}} \mathbf{H}_{2^{k-1}} \begin{bmatrix} f_{X,0} \\ f_{X,1} \\ \vdots \\ f_{X,2^{k-1}-1} \end{bmatrix} = \frac{1}{2^{k-1}} \begin{bmatrix} \mathbf{H}_{2^{k-1}} & \mathbf{0}_{2^{k-1} \times 2^{k-1}} \end{bmatrix} \mathbf{f}_{X,k+1},$$

$$\mathbf{f}_{k,1} = \frac{1}{2^{k-1}} \mathbf{H}_{2^{k-1}} \begin{bmatrix} f_{X,2^{k-1}} \\ f_{X,2^{k-1}+1} \\ \vdots \\ f_{X,2^k-1} \end{bmatrix} = \frac{1}{2^{k-1}} \begin{bmatrix} \mathbf{0}_{2^{k-1} \times 2^{k-1}} & \mathbf{H}_{2^{k-1}} \end{bmatrix} \mathbf{f}_{X,k+1}. \tag{82}$$

The following transformation is applied to $\mathbf{f}_{k,0}$ and $\mathbf{f}_{k,1}$:

$$\mathbf{f}_{k,+} = \frac{1}{2} [\mathbf{f}_{k,0} + \mathbf{f}_{k,1}], \quad \mathbf{f}_{k,-} = \frac{1}{2} [\mathbf{f}_{k,0} - \mathbf{f}_{k,1}]. \tag{83}$$

The inverse of (83) is

$$\mathbf{f}_{k,0} = \mathbf{f}_{k,+} + \mathbf{f}_{k,-}, \quad \mathbf{f}_{k,1} = \mathbf{f}_{k,+} - \mathbf{f}_{k,-}. \tag{84}$$

By applying (84) to (81), one obtains

$$\begin{aligned} z_{k+1,RF}(t) &= \Re \left\{ z_k^0(t) e^{2\pi j_k^T [\mathbf{f}_{k,+} + \mathbf{f}_{k,-}]t} + z_k^1(t) e^{2\pi j_k^T [\mathbf{f}_{k,+} - \mathbf{f}_{k,-}]t} \right\} \\ &= \Re \left\{ \left[z_k^0(t) e^{2\pi j_k^T \mathbf{f}_{k,-}t} + z_k^1(t) e^{-2\pi j_k^T \mathbf{f}_{k,-}t} \right] e^{2\pi j_k^T \mathbf{f}_{k,+}t} \right\}. \end{aligned} \tag{85}$$

The vector with the imaginary units j_k can be expressed as $j_0 \mathbf{D}_k \mathbf{k}_k$ using (24). Thus, the signal within square brackets in (85) can be written as

$$\begin{aligned} & z_k^0(t) e^{2\pi j_k^T \mathbf{f}_{k,-}t} + z_k^1(t) e^{-2\pi j_k^T \mathbf{f}_{k,-}t} \\ &= z_k^0(t) e^{2\pi j_0 \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t} + z_k^1(t) e^{-2\pi j_0 \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t} \\ &= z_k^0(t) \cos(2\pi \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t) + j_0 z_k^0(t) \sin(2\pi \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t) \\ &+ z_k^1(t) \cos(2\pi \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t) - j_0 z_k^1(t) \sin(2\pi \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t) \\ &= \left[z_k^0(t) + z_k^1(t) \right] \cos(2\pi \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t) + j_0 \left[z_k^0(t) - z_k^1(t) \right] \sin(2\pi \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t) \end{aligned} \tag{86}$$

In the last equation, trigonometric properties including the Euler’s formulas have been used [26]. In analogy to the proof provided for the bicomplex case, we can define a new signal in \mathbb{C}_{k+1} :

$$\begin{aligned} z_{k+1,I}(t) &= z_k^0(t) + z_k^1(t) \\ z_{k+1,Q}(t) &= j_0 \left[z_k^0(t) - z_k^1(t) \right] \\ z_{k+1}(t) &= z_{k+1,I}(t) - j_{2^{k-1}} z_{k+1,Q}(t) \in \mathbb{C}_{k+1}, \end{aligned} \tag{87}$$

where $j_{2^{k-1}}$ is the new base imaginary unit generating \mathbb{C}_{k+1} from \mathbb{C}_k . At the same time, it is possible to introduce the multicomplex exponential:

$$e^{2\pi j_{2^{k-1}} \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t} = \cos(2\pi \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t) + j_{2^{k-1}} \sin(2\pi \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t). \tag{88}$$

Using (87) and (88), it is finally possible to express (86) as

$$z_k^0(t) e^{2\pi j_k^T \mathbf{f}_{k,-}t} + z_k^1(t) e^{-2\pi j_k^T \mathbf{f}_{k,-}t} = \Re^{j_{2^{k-1}}} \left\{ z_{k+1}(t) e^{2\pi j_{2^{k-1}} \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-}t} \right\}. \tag{89}$$

In this case, $\Re^{j_{2^{k-1}}} \{ \cdot \}$ denotes the real part operation only with respect to the $j_{2^{k-1}}$ unit, i.e.,

$$\Re^{j_{2^{k-1}}} \{ z_{k+1} \} = \frac{1}{2} \left[z_{k+1} + z_{k+1}^{*j_{2^{k-1}}} \right] \tag{90}$$

where $z_{k+1} \in \mathbb{C}_{k+1}$ and $z_{k+1}^{*j_{2^{k-1}}}$ is its $j_{2^{k-1}}$ -conjugate obtained by negating the sign of the terms multiplying the $j_{2^{k-1}}$ unit. If (87) is considered, the real part of $z_{k+1}(t)$ with respect to $j_{2^{k-1}}$ is $z_{k+1,I}(t)$. Using these properties, (85) becomes

$$\begin{aligned}
 z_{k+1,RF}(t) &= \Re \left\{ z_{k+1}(t) e^{2\pi j_{2^{k-1}} \mathbf{k}_k^T \mathbf{D}_k^T \mathbf{f}_{k,-} t} e^{-2\pi j_k^T \mathbf{f}_{k,+} t} \right\} \\
 &= \Re \left\{ z_{k+1}(t) e^{2\pi \left[\begin{matrix} \mathbf{j}_k^T & j_{2^{k-1}} \mathbf{k}_k^T \end{matrix} \right] \left[\begin{matrix} \mathbf{f}_{k,+} \\ \mathbf{D}_k^T \mathbf{f}_{k,-} \end{matrix} \right]^T t} \right\} \\
 &= \Re \left\{ z_{k+1}(t) e^{2\pi j_{k+1}^T \mathbf{f}_{k+1} t} \right\},
 \end{aligned} \tag{91}$$

which completes the proof of (77). Equation (91) provides a way to compute \mathbf{f}_{k+1} , which is given by the following:

$$\begin{aligned}
 \mathbf{f}_{k+1} &= \begin{bmatrix} \mathbf{f}_{k,+} \\ \mathbf{D}_k^T \mathbf{f}_{k,-} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{2^{k-1}} & \mathbf{0}_{2^{k-1} \times 2^{k-1}} \\ \mathbf{0}_{2^{k-1} \times 2^{k-1}} & \mathbf{D}_k \end{bmatrix} \begin{bmatrix} \mathbf{f}_{k,+} \\ \mathbf{f}_{k,-} \end{bmatrix} \\
 &= \frac{1}{2} \begin{bmatrix} \mathbf{I}_{2^{k-1}} & \mathbf{I}_{2^{k-1}} \\ \mathbf{D}_k & -\mathbf{D}_k \end{bmatrix} \begin{bmatrix} \mathbf{f}_{k,0} \\ \mathbf{f}_{k,1} \end{bmatrix} \\
 &= \frac{1}{2^k} \begin{bmatrix} \mathbf{I}_{2^{k-1}} & \mathbf{I}_{2^{k-1}} \\ \mathbf{D}_k & -\mathbf{D}_k \end{bmatrix} \begin{bmatrix} \mathbf{H}_{2^{k-1}} & \mathbf{0}_{2^{k-1} \times 2^{k-1}} \\ \mathbf{0}_{2^{k-1} \times 2^{k-1}} & \mathbf{H}_{2^{k-1}} \end{bmatrix} \mathbf{f}_{X,k+1} \\
 &= \frac{1}{2^k} \begin{bmatrix} \mathbf{H}_{2^{k-1}} & \mathbf{H}_{2^{k-1}} \\ \mathbf{D}_k \mathbf{H}_{2^{k-1}} & -\mathbf{D}_k \mathbf{H}_{2^{k-1}} \end{bmatrix} \mathbf{f}_{X,k+1} = \frac{1}{2^k} \mathbf{H}_{2^k} \mathbf{f}_{X,k+1}.
 \end{aligned} \tag{92}$$

Equation (92) completes the proof of (78) and provides a way to recursively compute \mathbf{H}_{2^k} . This construction coincides with (56). Moreover, (78) is analogous to (69), where phase vectors have been replaced by frequency vectors.

Formula (77) provides an alternative way to describe a modulation process involving signals from several frequencies. Consider, for instance, the signals broadcast by GNSS. Galileo, the European GNSS, currently broadcasts Open Service (OS) signals on four different frequencies [27]. These signals are transmitted synchronously and coherently and can be jointly used to estimate the user position. A summary of the Galileo frequency plan is provided in the top part of Figure 1, which also describes the associated RF up-conversion process according to multicomplex modulation formula (77).

Galileo satellites currently broadcast signals into the E5a (1176.45 MHz), E5b (1207.14 MHz), E6 (1278.75 MHz) and E1 (1575.42 MHz) frequency bands. Different modulations are adopted including Binary Phase Shifting Keying (BPSK) and Binary Offset Carrier (BOC) signals. The four components depicted in the top part of Figure 1 are schematic representations of the spectra of the four Galileo OS signals. These four components can be thought of as a single baseband multicomplex signal $\in \mathbb{C}_3$ modulated through (77) to different RFs. This modulation process is obtained considering the frequencies computed using (92). The four components are first brought to a common RF, which corresponds to the average of the four final signal RFs. This common frequency is obtained by considering the first row of the Hadamard matrix, \mathbf{H}_4 . The other frequencies obtained from (92) progressively split the four signal components on different RFs. This modulation model can be used for the design of advanced demodulation processes as demonstrated in [21] for bicomplex signals: having a single multicomplex model allows one to jointly use all available components to estimate common frequency terms.

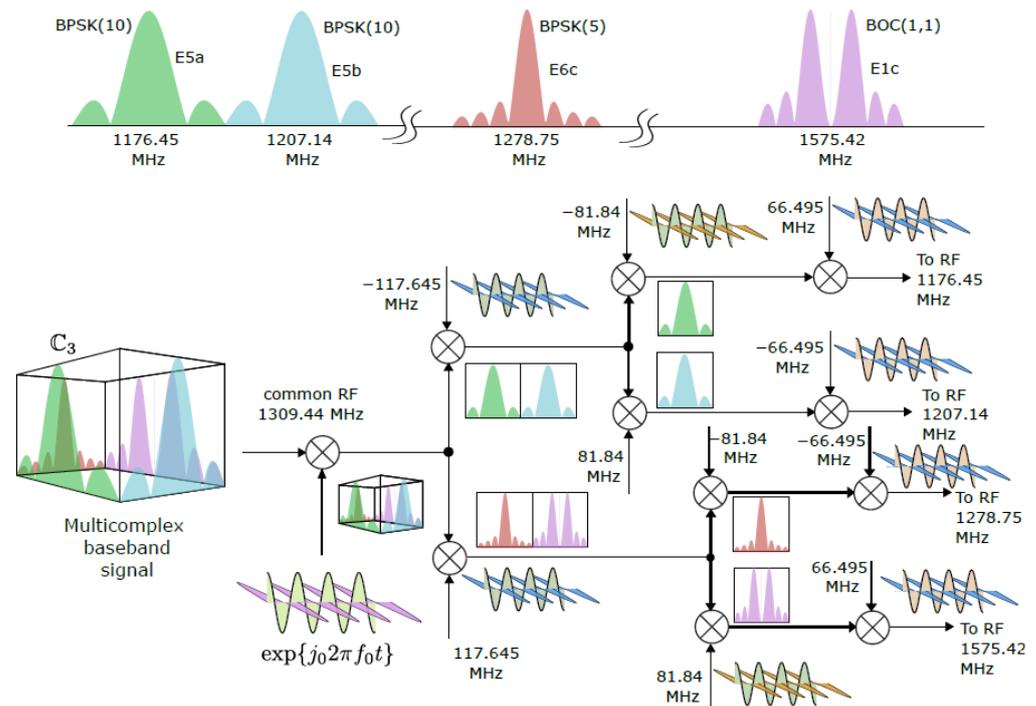


Figure 1. Schematic representation of the Galileo OS frequency plan (top part) and representation of the associated RF up-conversion process as a multicomplex modulation.

Orthogonal Decomposition

Using the set defined by (39), it is possible to derive the orthogonal decomposition of the multicomplex baseband signal $z_n(t)$. In particular,

$$z_n(t) = 2^{n-1} \sum_{i=0}^{2^{n-1}-1} x_i(t) e_{n,i} \tag{93}$$

where $x_i(t)$ is the baseband version of the i th RF signal $x_{i,RF}(t)$ introduced in (79). Also, in this case, the proof is done by induction. A proof for the induction base can be found in [21] for bicomplex numbers. In order to prove the induction step, one assumes (93) is true for $n - 1$ and proves its validity for n . From (87), it follows

$$\begin{aligned} z_n(t) &= z_{n,I}(t) - j_{2^{n-2}} z_{n,Q}(t) \\ &= [z_{n-1}^0(t) + z_{n-1}^1(t)] - j_0 j_{2^{n-2}} [z_{n-1}^0(t) - z_{n-1}^1(t)] \\ &= 2z_{n-1}^0(t) \frac{1 - k_{2^{n-2}}}{2} + 2z_{n-1}^1(t) \frac{1 + k_{2^{n-2}}}{2}. \end{aligned} \tag{94}$$

By applying the induction hypothesis to both $z_{n-1}^0(t)$ and $z_{n-1}^1(t)$, one has

$$\begin{aligned} z_{n-1}^0(t) &= 2^{n-2} \sum_{i=0}^{2^{n-2}-1} x_i(t) e_{n-1,i} \\ z_{n-1}^1(t) &= 2^{n-2} \sum_{i=0}^{2^{n-2}-1} x_{i+2^{n-2}}(t) e_{n-1,i} \end{aligned} \tag{95}$$

where $x_i(t)$ and $x_{i+2^{n-2}}(t)$ are the baseband versions of the RF signals introduced in (79). The proof of the induction step is finally obtained by combining (94) and (95) and exploiting recursive relationships (41).

Equation (93) provides a simple way to determine $z_n(t)$ from the baseband complex version of the original RF signals, $x_{i,RF}(t)$. This process is illustrated in Figure 2, which shows the scatter plots of four baseband Quadrature Phase Shift Keying (QPSK) signals affected by noise. The signals are sampled, and each point in the figure represents a single data symbol. This type of modulation is widely used in many communication and navigation systems.

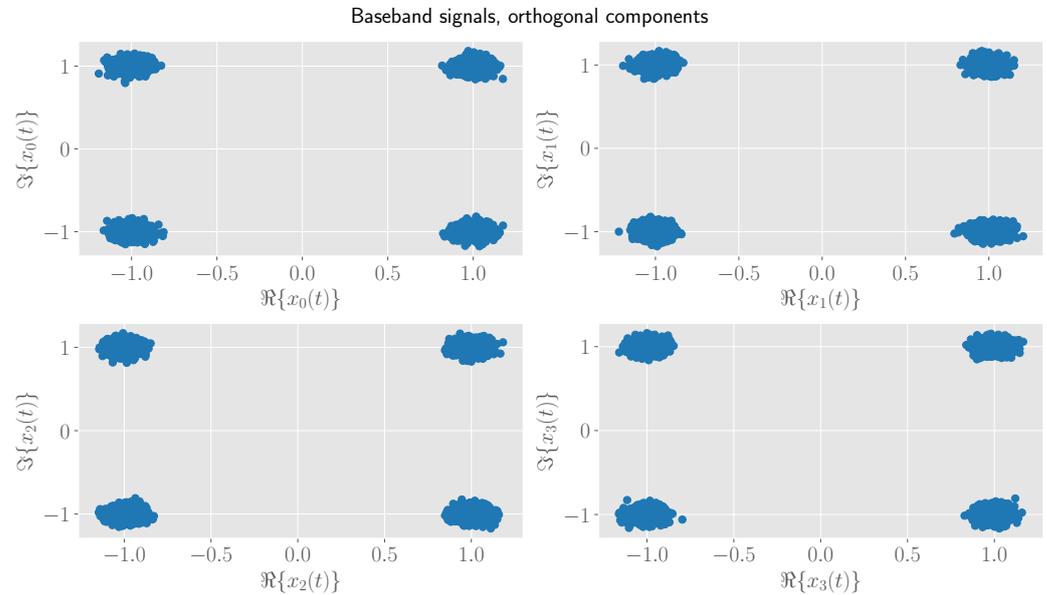


Figure 2. Scatter plots of four baseband QPSK signals affected by noise. The signals are used as orthogonal components to form a single multicomplex signal $\in \mathbb{C}_3$ according to (93).

The signals are used as orthogonal components to form $z_n(t)$ according to (93): the resulting signal is shown in Figure 3. $z_n(t)$ is expressed as $\mathbf{k}_3^T \mathbf{z}_M(t)$ with $\mathbf{z}_M(t)$ a four-dimensional complex vector: each box in Figure 3 represents a different component of $\mathbf{z}_M(t)$. These components are obtained through a Hadamard transform, and the QPSK modulations depicted in Figure 2 are mapped to higher-order Quadrature Amplitude Modulation (QAM) modulations.

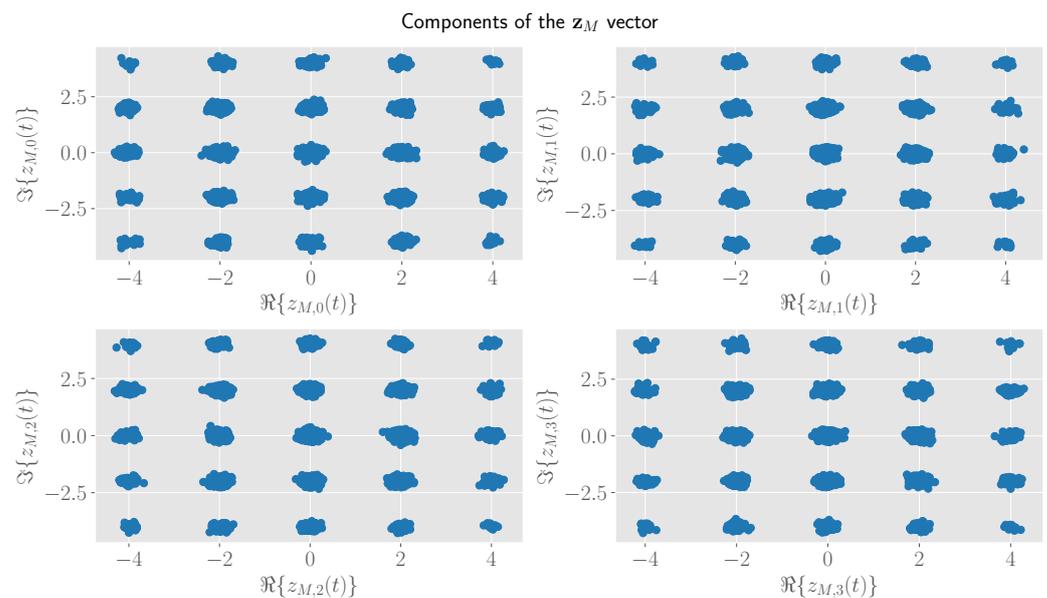


Figure 3. Scatter plots of the four complex components of $z_n(t)$. $z_n(t)$ is expressed as $\mathbf{k}_3^T \mathbf{z}_M(t)$ with $\mathbf{z}_M(t)$ a four-dimensional complex vector. Each subplot represents a different component of $\mathbf{z}_M(t)$.

Note that the symbols shown in the four boxes of Figure 3 have different probabilities to occur. This happens even if the symbols in the original QPSK modulations were selected with uniform probabilities.

The four real components of the hyperbolic modulus of the multicomplex signal depicted in Figure 3 are provided in Figure 4 as a function of time. From the figure, it emerges that the signal energy is concentrated on the real component of the hyperbolic modulus, whereas the other components are zero mean and contain only noise.

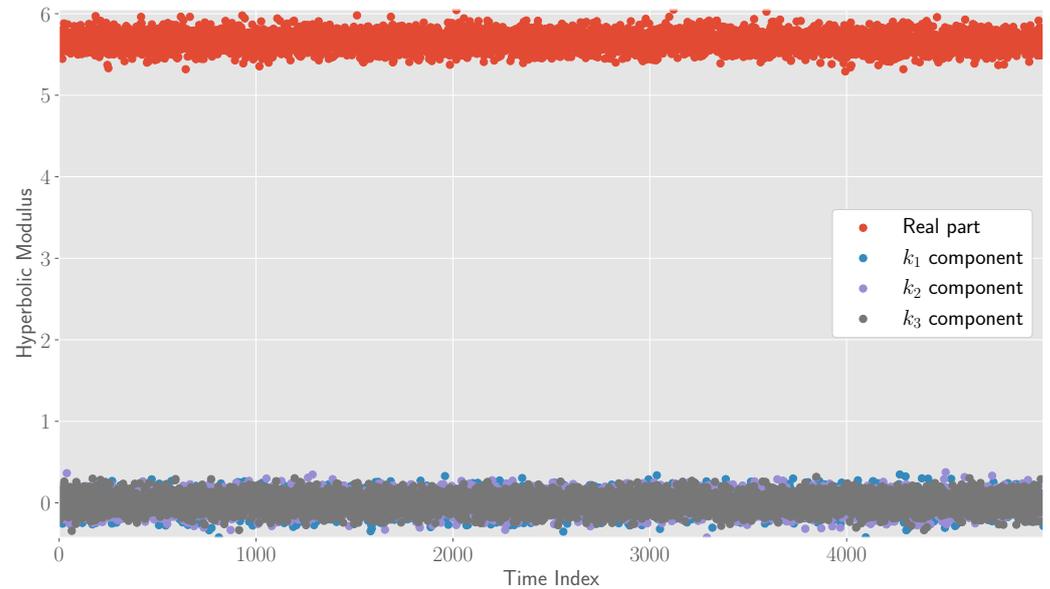


Figure 4. Hyperbolic modulus of the multicomplex signal obtained in Figure 3. The four real components of the hyperbolic modulus are depicted separately as a function of time.

The real part of the hyperbolic modulus is equal to the average of the absolute values of the orthogonal components. In this example, the orthogonal components are the four QPSK signals multiplied by four (that is 2^{3-1}). If noise is neglected, the symbols in Figure 2 are all of the form $\pm 1 \pm j_0$ and have all absolute values equal to $\sqrt{2}$. Thus, the real part of the hyperbolic modulus in Figure 4 assumes values close to $4\sqrt{2} \approx 5.66$. The other components, which multiply the hyperbolic units, k_1 , k_2 and k_3 , are close to zero as a result of the mixing effect of the Hadamard transform. In [21], it was shown that communication parameters required for the reception of bicomplex signals can be estimated through the maximization of the real part of their hyperbolic modulus. Indeed, when properly aligned in phase, the components multiplying the hyperbolic units should be minimized.

6. Discussion

In this paper, multicomplex numbers were first reviewed, and a representation based on vectors with pure imaginary and hyperbolic units was proposed. Orthogonal decomposition and polar representation were also considered. Properties of multicomplex numbers have been used to derive a generalized baseband signal representation formula. In particular, it was shown that a set of 2^{n-1} RF signals admits a single multicomplex baseband representation. The sum of the RF signals is indeed given by the real part of the product of a multicomplex baseband signal and a multicomplex exponential that is used to up-convert the different components to different RFs. An orthogonal decomposition formula for the multicomplex baseband signal was also provided. The frequencies appearing in the multicomplex exponential used for the up-conversion process are obtained as the Hadamard transform of the original signal RFs. This is the same relationship existing between the phases of the polar representation of a multicomplex number and the phases of its orthogonal components. Thus, the obtained polar representation can be used for the design of multi-dimensional Phase Lock Loops (PLLs) for the recovery and down-conversion of

multiple RF signals. The multicomplex representation obtained will be used for the design of multi-frequency acquisition and tracking algorithms jointly processing several signals from different RFs.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Ell, T.A.; Sangwine, S.J. Hypercomplex Fourier Transforms of Color Images. *IEEE Trans. Image Process.* **2007**, *16*, 22–35. [[CrossRef](#)]
2. Snopek, K.M. The n-D analytic signals and Fourier spectra in complex and hypercomplex domains. In Proceedings of the International Conference on Telecommunications and Signal Processing (TSP), Budapest, Hungary, 18–20 August 2011; pp. 423–427. [[CrossRef](#)]
3. Alfsmann, D.; Göckler, H.G.; Sangwine, S.J.; Ell, T.A. Hypercomplex algebras in digital signal processing: Benefits and drawbacks. In Proceedings of the European Signal Processing Conference (EUSIPCO), Poznan, Poland, 3–7 September 2007; pp. 1322–1326.
4. Valkova-Jarvis, Z.; Poulkov, V.; Stoyanov, V.; Mihaylova, D.; Iliev, G. A Method for the Design of Bicomplex Orthogonal DSP Algorithms for Applications in Intelligent Radio Access Networks. *Symmetry* **2022**, *14*, 613. [[CrossRef](#)]
5. Mengüç, E.C.; Acir, N.; Mandic, D.P. A Class of Online Censoring Based Quaternion-Valued Least Mean Square Algorithms. *IEEE Signal Process. Lett.* **2023**, *30*, 244–248. [[CrossRef](#)]
6. Lin, D.; Zhang, Q.; Chen, S.; Wang, S. The Generalized HR q -Derivative and Its Application to Quaternion Least Mean Square Algorithm. *IEEE Signal Process. Lett.* **2022**, *29*, 857–861. [[CrossRef](#)]
7. Took, C.C.; Mandic, D.P. Quaternion-Valued Stochastic Gradient-Based Adaptive IIR Filtering. *IEEE Trans. Signal Process.* **2010**, *58*, 3895–3901. [[CrossRef](#)]
8. Alpay, D.; Diki, K.; Vajiac, M. Two Bicomplex and One Multicomplex Least Mean Square algorithms. *arXiv* **2023**, arXiv:cs.LG/2209.11899.
9. El-Melegy, M.T.; Kamal, A.T. Linear Regression Classification in the Quaternion and Reduced Biquaternion Domains. *IEEE Signal Process. Lett.* **2022**, *29*, 469–473. [[CrossRef](#)]
10. Altmann, S.L. *Rotations, Quaternions, and Double Groups*; Dover Publications: Mineola, NY, USA, 2005.
11. Ell, T.A.; Le Bihan, N.; Sangwine, S.J. *Quaternion Fourier Transforms for Signal and Image Processing*; Digital Signal and Image Processing; Wiley: Hoboken, NJ, USA; iSTE: London, UK, 2014.
12. Alfsmann, D. On families of 2^N -dimensional hypercomplex algebras suitable for digital signal processing. In Proceedings of the European Signal Processing Conference (EUSIPCO), Florence, Italy, 4–8 September 2006; pp. 1–4.
13. Richter, W.D. On Complex Numbers in Higher Dimensions. *Axioms* **2022**, *11*, 22. [[CrossRef](#)]
14. Catoni, F.; Cannata, R.; Zampetti, P. An Introduction to Commutative Quaternions. *Adv. Appl. Clifford Algebr.* **2006**, *16*, 1–28. [[CrossRef](#)]
15. Pei, S.C.; Chang, J.H.; Ding, J.J. Commutative reduced biquaternions and their Fourier transform for signal and image processing applications. *IEEE Trans. Signal Process.* **2004**, *52*, 2012–2031. [[CrossRef](#)]
16. Alpay, D.; Luna-Elizarrarás, M.E.; Shapiro, M.; Struppa, D.C. *Basics of Functional Analysis with Bicomplex Scalars, and Bicomplex Schur Analysis*; SpringerBriefs in Mathematics; Springer: Cham, Switzerland, 2014.
17. Price, G.B. *An Introduction to Multicomplex Spaces and Functions*, 1st ed.; CRC Press: Boca Raton, FL, USA, 1991. [[CrossRef](#)]
18. Navarro-Moreno, J.; Fernández-Alcalá, R.M.; Jiménez-López, J.D.; Ruiz-Molina, J.C. Tessarine signal processing under the T-properness condition. *J. Frankl. Inst.* **2020**, *357*, 10100–10126. [[CrossRef](#)]
19. Segre, C. Le rappresentazioni reali delle forme complesse e gli enti iperalgebrici. *Math. Ann.* **1891**, *40*, 413–467. [[CrossRef](#)]
20. Alfsmann, D.; Göckler, H.G. On hyperbolic complex LTI digital systems. In Proceedings of the European Signal Processing Conference (EUSIPCO), Poznań, Poland, 3–7 September 2007; pp. 1332–1336.
21. Borio, D. Bicomplex Representation and Processing of GNSS Signals. *NAVIGATION J. Inst. Navig.* **2023**, *70*, 1–33. [[CrossRef](#)]
22. Fjelstad, P.; Gal, S.G. n-Dimensional hyperbolic complex numbers. *Adv. Appl. Clifford Algebr.* **1998**, *8*, 47–68. [[CrossRef](#)]
23. Yarlagadda, R.K.R.; Hershey, J.E. *Hadamard Matrix Analysis and Synthesis*; The Springer International Series in Engineering and Computer Science; Springer: New York, NY, USA, 1997. [[CrossRef](#)]
24. Proakis, J.G.; Salehi, M. *Communication Systems Engineering*, 2nd ed.; Prentice-Hall: Upper Saddle River, NJ, USA, 2001.
25. Casado, J.M.V.; Hewson, R. Algorithm 1008: Multicomplex Number Class for Matlab, with a Focus on the Accurate Calculation of Small Imaginary Terms for Multicomplex Step Sensitivity Calculations. *ACM Trans. Math. Softw.* **2020**, *46*, 1–26. [[CrossRef](#)]

-
26. Nasiruzzaman, M.; Mursaleen, M. The Multicomplex Numbers and Their Properties on Some Elementary Functions. *Thai J. Math.* **2023**, *21*, 77–100.
 27. European Union. Galileo Open Service Signal-In-Space Interface Control Document (OS SIS ICD). In *Technical Report 2.1*; Publications Office of the European Union: Luxembourg, 2023. [[CrossRef](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.