A Distributed Quaternion Kalman Filter With Applications to Smart Grid and Target Tracking

Sayed Pouria Talebi, Sithan Kanna, *Student Member, IEEE*, and Danilo P. Mandic, *Fellow, IEEE*

***Abstract*—Recent advances in sensor and communication technologies have made the deployment of sensor networks in a variety of roles feasible, including smart grid management applications and collaborative target tracking solutions. While most research in distributed adaptive signal processing is conducted in the real and complex domains, inherently in many real-world applications the data sources are three-dimensional. This scenario is ideally suited for quaternions in terms of both convenience of representation and mathematical tractability. In this paper, we expand the concept of distributed Kalman filtering to the quaternion domain in order to develop a robust distributed quaternion Kalman filtering algorithm for data fusion over sensor networks dealing with three-dimensional data. For rigor, the mean and mean square behavior of the algorithm are analyzed. Finally, the developed algorithm is used to estimate the nominal system frequency in power distribution networks and for collaborative target tracking applications.**

***Index Terms*—Distributed estimation, frequency estimation, smart grid, target tracking, quaternion-valued signal processing.**

I. Introduction

In RECENT years, sensor networks have been used in a variety of applications such as collaborative target tracking, distributed fault detection, control of unmanned aerial vehicles, and automated vehicle guidance technology [1]–[18]. In these applications, algorithms based on Kalman filtering have proven to be advantageous in terms of enhanced accuracy and faster convergence rates, due to their underlying state space model that accounts for observational noise. In addition, owing to the low implementation cost and computational efficiency that distributed estimation and tracking techniques offer, as compared to their centralized counterparts, distributed signal processing algorithms have proven to be computationally efficient, scalable with the size of the network, robust to link failure, and suitable for real-time implementation [8]–[12], [18], [19]. Most distributed signal processing algorithms are developed in the real and complex domains; however, in our three-dimensional world, real and complex-valued models lack the dimensionality necessary to adequately represent the signal of interest.

Manuscript received January 30, 2016; revised July 20, 2016; accepted September 19, 2016. Date of publication October 18, 2016; date of current version November 4, 2016. The guest editor coordinating the review of this manuscript and approving it for publication was Prof. Cédric Richard.

The authors are with the Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, U.K. (e-mail: s.talebi12@imperial.ac.uk; shri.kanagasabapathy08@imperial.ac.uk; d.mandic@imperial.ac.uk).

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/TSIPN.2016.2618321

Quaternions have been used in mathematics, physics, and computer graphics in order to model three-dimensional rotations and orientation in a compact and computationally efficient fashion, where their division algebra allows us to avoid problems associated with rotation, e.g. gimbal lock [20]. Quaternions provide accurate and mathematically tractable solutions with fewer constraints than those obtained in the real domain through vector algebras. In addition, the introduction of the $\mathbb{HR}$-calculus [21], [22], a framework for calculating the derivatives of quaternion-valued functions, and the augmented second-order statistics of quaternion-valued random signals [23], [25], a framework for exploiting the full second-order statistical information of quaternion-valued random signals, have led to the development of quaternion-valued signal processing algorithms, such as the class of quaternion Kalman filters developed in [26]. These are shown to outperform their real and complex-valued counterparts in applications including frequency estimation in smart grids [27], color image processing [28], [29], bearings-only-tracking [26], spacecraft orientation tracking [30], kernel learning [31], [32], and wind profile forecasting [33]. Although a diffusion quaternion least mean square algorithm does exist [34], a fully distributed quaternion-valued sequential state estimator is still lacking.

In light of the advantages that quaternion-valued signal processing algorithms offer, we expand the framework of quaternion-valued Kalman filtering to the distributed setting in order to develop a rigorous distributed quaternion Kalman filter applicable for frequency estimation in three-phase power distribution networks and collaborative target tracking. Quaternions offer the dimensionality necessary to model such signals directly in the multi-dimensional domain where they live. The distributed quaternion Kalman filter is developed through decomposing the operations of the centralized quaternion Kalman filter in such a way that they can be performed locally by the individual nodes (sensors) of the network. The performance analysis of the developed algorithm shows that it is unbiased; moreover, in order to quantify the mean square behavior of the developed algorithm, a recursive expression for the augmented covariance matrix of the estimation error is derived. This also allows for the concept to be expanded to multi-task settings through the introduction of a confidence measure.

Mathematical notations: Scalars, column vectors, and matrices are represented by lowercase, bold lowercase, and bold uppercase letters. The augmented state vector at time instant n is denoted by $\mathbf{x}_n^a$, while $\mathbf{I}$ represents the identity matrix with the same number of rows as the augmented state vector. The transpose, Hermitian transpose, and trace operators are denoted by $(\cdot)^T$, $(\cdot)^H$, and $\mathrm{Tr}(\cdot)$, whereas $E[\cdot]$ denotes the statistical expectation operator. The Kronecker product is denoted by $\otimes$ and

the operator Vec$(\cdot)$ transforms a matrix into a column vector by stacking its columns. Finally, the real and quaternion domains are denoted by $\mathbb{R}$ and $\mathbb{H}$.

II. Background

The skew-field of quaternions is a four-dimensional, non-commutative, associative, division algebra. A quaternion variable $q \in \mathbb{H}$ consists of a real part, $\Re(q)$, and a three-dimensional imaginary part or pure quaternion, $\Im(q)$, which comprises three components $\Im_i(q)$, $\Im_j(q)$, and $\Im_k(q)$; hence, a variable $q \in \mathbb{H}$ can be expressed as

$$\begin{aligned} q = \Re(q) + \Im(q) &= \Re(q) + \Im_i(q) + \Im_j(q) + \Im_k(g) \\ &= q_r + iq_i + jq_j + kq_k \end{aligned}$$

where $q_r, q_i, q_j, q_k \in \mathbb{R}$. The unit vectors i, j, and k form the orthonormal basis for the quaternion imaginary subspace and obey the following product rules

$$\begin{gathered} ij = -ji = k, jk = -kj = i, ki = -ik = j, \\ i^2 = j^2 = k^2 = ijk = -1 \end{gathered}$$

which also illustrate the non-commutativity property.

The involution of $q \in \mathbb{H}$ around $\mu \in \mathbb{H}$ is defined as $q^\mu \triangleq \mu q \mu^{-1}$ [35]. In particular, the three self-inversing involutions

$$\begin{aligned} q^i &= -iqi = q_r + iq_i - jq_j - kq_k \\ q^j &= -jqj = q_r - iq_i + jq_j - kq_k \\ q^k &= -kqk = q_r - iq_i - jq_j + kq_k \end{aligned}$$

that rotate the imaginary part of q by an angle of π around the i-, j-, and k-axis [35], are seen as the quaternion equivalent of the complex conjugate operator. The real-valued components of a quaternion number, $q \in \mathbb{H}$, can be expressed using these involutions as [22]–[27], [31], [33], [34]

$$\begin{aligned} q_r &= \frac{1}{4}\left(q + q^i + q^j + q^k\right) \quad q_i = \frac{1}{4i}\left(q + q^i - q^j - q^k\right) \\ q_j &= \frac{1}{4j}\left(q - q^i + q^j - q^k\right) \quad q_k = \frac{1}{4k}\left(q - q^i - q^j + q^k\right). \end{aligned} \tag{1}$$

Furthermore, the quaternion conjugate is also a self-inverse involution that rotates the imaginary part of q around the i-, j-, and k-axis simultaneously and is defined as [25], [35]

$$q^* = \Re(q) - \Im(q) = \frac{1}{2}\left(q^i + q^j + q^k - q\right)$$

while the norm of $q \in \mathbb{H}$ is given by

$$|q| = \sqrt{qq^*} = \sqrt{q_r^2 + q_i^2 + q_j^2 + q_k^2}.$$

A quaternion $q \in \mathbb{H}$ can alternatively be expressed by its polar presentation, given by [36]

$$q = |q|e^{\xi\theta} = |q|\big(\cos(\theta) + \xi\sin(\theta)\big)$$

where

$$\xi = \frac{\Im(q)}{|\Im(q)|}, \quad \theta = \operatorname{atan}\left(\frac{|\Im(q)|}{\Re(q)}\right).$$

Moreover, it is straightforward to prove that the $\sin(\cdot)$ and $\cos(\cdot)$ functions can be expressed as

$$\sin(\theta) = \frac{1}{2\xi}\left(e^{\xi\theta} - e^{-\xi\theta}\right), \ \cos(\theta) = \frac{1}{2}\left(e^{\xi\theta} + e^{-\xi\theta}\right) \tag{2}$$

where[1] $\xi^2 = -1$.

Consider the quaternion-valued function $f(\cdot) : \mathbb{H}^N \to \mathbb{H}$ and the parameter vector $\mathbf{q} \in \mathbb{H}^N$; then, the function f is differentiable with respect to $\mathbf{q}$ if and only if it satisfies the Cauchy-Riemann-Fueter condition given by given by [22], [37]

$$\frac{\partial f(\mathbf{q})}{\partial \mathbf{q}^*} = \frac{1}{4}\Big(\frac{\partial f}{\partial \mathbf{q}_r} + i\frac{\partial f}{\partial \mathbf{q}_i} + j\frac{\partial f}{\partial \mathbf{q}_j} + k\frac{\partial f}{\partial \mathbf{q}_k}\Big) = 0.$$

However, the Cauchy-Riemann-Fueter condition imposes a severe restriction on differentiable quaternion-valued functions, only allowing for the differentiation of linear functions. One elegant solution to this problem is the $\mathbb{HR}$-calculus [21], [22], which based on the expressions in (1), establishes a duality between $\mathbb{R}^4$ and $\mathbb{H}$. A quaternion function, $f(\mathbf{q} = \mathbf{q}_r + i\mathbf{q}_i + j\mathbf{q}_j + k\mathbf{q}_k) : \mathbb{H}^N \to \mathbb{H}$ can now be expressed in terms of the orthogonal quaternion basis $\mathbf{q}$, $\mathbf{q}^i$, $\mathbf{q}^j$, and $\mathbf{q}^k$, such that $f(\mathbf{q}^a = [\mathbf{q}, \mathbf{q}^i, \mathbf{q}^j, \mathbf{q}^k]^T) : \mathbb{H}^{4N} \to \mathbb{H}$, where $\mathbf{q}^a$ is referred to as the augmented quaternion vector. Then, by considering the real-valued components of $f(\mathbf{q}^a) = f_r(\mathbf{q}^a) + if_i(\mathbf{q}^a) + jf_j(\mathbf{q}^a) + kf_k(\mathbf{q}^a)$ and through exploiting the isomorphism between $\mathbb{R}^4$ and $\mathbb{H}$, a relation can be established between the derivatives taken in $\mathbb{R}^4$ and those taken directly in $\mathbb{H}$, allowing for a unified framework for calculating the derivatives and establishing the gradients of quaternion-valued functions directly in the quaternion domain.

The isomorphism between the augmented quaternion vector $\mathbf{q}^a = [\mathbf{q}, \mathbf{q}^i, \mathbf{q}^j, \mathbf{q}^k]^T \in \mathbb{H}^{4N}$ and $[\mathbf{q}_r, \mathbf{q}_i, \mathbf{q}_j, \mathbf{q}_k]^T \in \mathbb{R}^{4N}$ has been instrumental in the development of the augmented quaternion statistics that allow for the full second-order statistical description of quaternion random variables through the use of the augmented covariance matrix [23]–[25], given by

$$\begin{aligned} \mathbf{C}_{\mathbf{q}^a} &= E[\mathbf{q}^a \mathbf{q}^{aH}] \\ &= \begin{bmatrix} \mathbf{C}_{\mathbf{q}\mathbf{q}} & \mathbf{C}_{\mathbf{q}\mathbf{q}^i} & \mathbf{C}_{\mathbf{q}\mathbf{q}^j} & \mathbf{C}_{\mathbf{q}\mathbf{q}^k} \\ \mathbf{C}_{\mathbf{q}^i\mathbf{q}} & \mathbf{C}_{\mathbf{q}^i\mathbf{q}^i} & \mathbf{C}_{\mathbf{q}^i\mathbf{q}^j} & \mathbf{C}_{\mathbf{q}^i\mathbf{q}^k} \\ \mathbf{C}_{\mathbf{q}^j\mathbf{q}} & \mathbf{C}_{\mathbf{q}^j\mathbf{q}^i} & \mathbf{C}_{\mathbf{q}^j\mathbf{q}^j} & \mathbf{C}_{\mathbf{q}^j\mathbf{q}^k} \\ \mathbf{C}_{\mathbf{q}^k\mathbf{q}} & \mathbf{C}_{\mathbf{q}^k\mathbf{q}^i} & \mathbf{C}_{\mathbf{q}^k\mathbf{q}^j} & \mathbf{C}_{\mathbf{q}^k\mathbf{q}^k} \end{bmatrix} \end{aligned} \tag{3}$$

where $\forall \zeta, \zeta' \in \{1, i, j, k\}, \mathbf{C}_{\mathbf{q}^\zeta\mathbf{q}^{\zeta'}} = E\big[\mathbf{q}^\zeta \mathbf{q}^{\zeta' H}\big]$.

Remark 1: All elements of the augmented covariance matrix in (3) are different involutions of $\mathbf{C}_{\mathbf{q}\mathbf{q}^i}$, $\mathbf{C}_{\mathbf{q}\mathbf{q}^j}$, $\mathbf{C}_{\mathbf{q}\mathbf{q}^k}$, and $\mathbf{C}_{\mathbf{q}\mathbf{q}}$. Therefore, the complete second-order information within the augmented covariance matrix is contained in the standard covariance, $\mathbf{C}_{\mathbf{q}\mathbf{q}}$, and the pseudo-covariances, $\mathbf{C}_{\mathbf{q}\mathbf{q}^i}$, $\mathbf{C}_{\mathbf{q}\mathbf{q}^j}$, and $\mathbf{C}_{\mathbf{q}\mathbf{q}^k}$.

To illustrate the need for such an approach, consider the minimum mean square error (MMSE) estimator of a variable, y, conditional to the observation, x, given by $\hat{y} = E[y|x]$. For

[1]Note that in order to express the $\sin(\cdot)$ and $\cos(\cdot)$ functions in their polar from, as in (2), ξ can be replaced with an arbitrary normalized pure quaternion number [36].

real-valued, zero-mean, and jointly Gaussian x and y, the solution is the standard linear estimator in the form of $\hat{y} = \mathbf{g}^T\mathbf{x}$, where $\mathbf{g} = [g_1, g_2, \ldots, g_N]^T$ is a vector of coefficients and $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T$ is a regressor vector of past observations. However, in the quaternion domain, following the complex domain analogy [38], [39], the MMSE estimator has to be expressed according to the individual components of the quaternion random variables; thus, for quaternion-valued y and x the MMSE estimator is given by

$$\hat{y} = E[y_r|x_r, x_i, x_j, x_k] + iE[y_i|x_r, x_i, x_j, x_k] + jE[y_j|x_r, x_i, x_j, x_k] + kE[y_k|x_r, x_i, x_j, x_k].$$

The expressions in (1) are now exploited to replace x_r, x_i, x_j, and x_k, to give

$$\hat{y} = E[y_r|x, x^i, x^j, x^k] + iE[y_i|x, x^i, x^j, x^k] + jE[y_j|x, x^i, x^j, x^k] + kE[y_k|x, x^i, x^j, x^k].$$

Therefore, for quaternion-valued, zero-mean, and jointly Gaussian x and y, the MMSE solution is in the form of a widely-linear estimator given by

$$\hat{y} = \mathbf{g}^T\mathbf{x} + \mathbf{h}^T\mathbf{x}^i + \mathbf{u}^T\mathbf{x}^j + \mathbf{v}^T\mathbf{x}^k$$

where $\mathbf{g}^T$, $\mathbf{h}^T$, $\mathbf{u}^T$, and $\mathbf{v}^T$ are quaternion-valued coefficient vectors and $\mathbf{x}$ is the regressor vector.

The augmented quaternion statistics in conjunction with the $\mathbb{HR}$-calculus have led to the development of a class of quaternion Kalman filters [26] that operate akin to their complex-valued counterparts. For example, consider the evolution of the quaternion-valued augmented state vector sequence $\{\mathbf{x}_n^a, n = 0, 1, 2, \ldots\}$, given by

$$\mathbf{x}_n^a = f_n(\mathbf{x}_{n-1}^a) + \boldsymbol{\nu}_n^a$$

where $f_n(\cdot)$ is the state evolution function at time instant n and $\{\boldsymbol{\nu}_n^a, n = 0, 1, 2, \ldots\}$ is the augmented state transition noise sequence. The objective is to track $\mathbf{x}_n^a$ in real-time through observations

$$\mathbf{y}_n^a = h_n(\mathbf{x}_n^a) + \boldsymbol{\omega}_n^a$$

where $\mathbf{y}_n^a$ and $h_n(\cdot)$ are respectively the augmented observation vector and observation function at time instant n, while $\{\boldsymbol{\omega}_n^a, n = 0, 1, 2, \cdots\}$ is the augmented measurement noise sequence. In order to simplify the analysis, we shall approximate the observation and state evolution functions in a widely-linear fashion as $f_n(\mathbf{x}_n^a) \simeq \mathbf{A}_n^a\mathbf{x}_n^a$ and $h_n(\mathbf{x}_n^a) \simeq \mathbf{H}_n^a\mathbf{x}_n^a$, where $\mathbf{A}_n^a$ and $\mathbf{H}_n^a$ are the Jacobian matrices of $f_n(\cdot)$ and $h_n(\cdot)$. Now, the augmented sate vector sequence can be tracked using the quaternion Kalman filter (QKF) given in its information formulation in Algorithm 1, where $\mathbf{C}_{\boldsymbol{\nu}_n^a}$ and $\mathbf{C}_{\boldsymbol{\omega}_n^a}$ denote the augmented covariance matrices of $\boldsymbol{\nu}_n^a$ and $\boldsymbol{\omega}_n^a$, while $\hat{\mathbf{x}}_{n|n-1}^a$ and $\hat{\mathbf{x}}_{n|n}^a$ represent the *a priori* and *a posteriori* estimates of $\mathbf{x}_n^a$.

Algorithm 1: Quaternion Kalman Filter (QKF) [26].

Initialize with:

$$\hat{\mathbf{x}}_{0|0}^a = E[\mathbf{x}_0^a]$$

$$\hat{\mathbf{M}}_{0|0}^a = E\big[(\mathbf{x}_0^a - E[\mathbf{x}_0^a])(\mathbf{x}_0^a - E[\mathbf{x}_0^a])^H\big]$$

Model update:

$$\hat{\mathbf{x}}_{n|n-1}^a = \mathbf{A}_n^a\hat{\mathbf{x}}_{n-1|n-1}^a$$

$$\hat{\mathbf{M}}_{n|n-1}^a = \mathbf{A}_n^a\hat{\mathbf{M}}_{n-1|n-1}^a\mathbf{A}_n^{aH} + \mathbf{C}_{\boldsymbol{\nu}_n^a}$$

Measurement update:

$$\hat{\mathbf{M}}_{n|n}^{a^{-1}} = \hat{\mathbf{M}}_{n|n-1}^{a^{-1}} + \mathbf{H}_n^{aH}\mathbf{C}_{\boldsymbol{\omega}_n^a}^{-1}\mathbf{H}_n^a$$

$$\mathbf{G}_n = \hat{\mathbf{M}}_{n|n}^a\mathbf{H}_n^{aH}\mathbf{C}_{\boldsymbol{\omega}_n^a}^{-1}$$

$$\hat{\mathbf{x}}_{n|n}^a = \hat{\mathbf{x}}_{n|n-1}^a + \mathbf{G}_n\big(\mathbf{y}_n^a - \mathbf{H}_n^a\hat{\mathbf{x}}_{n|n-1}^a\big)$$

III. The Distributed Quaternion Kalman Filter

Consider a set of sensors denoted by $\mathcal{N}$ that are interconnected in a network and let the neighborhood of a node be the subset of nodes that communicate with that node, including self-communication. Organizing all observations made by different nodes throughout the network in the column vector

$$\mathbf{y}_{col,n}^a = [\mathbf{y}_{1,n}^{aT}, \ldots, \mathbf{y}_{|\mathcal{N}|,n}^{aT}]^T$$

where $\mathbf{y}_{m,n}^a$ represents the augmented observation vector at node m at time n and $|\mathcal{N}|$ denotes the number of nodes in the network, allows the augmented state vector sequence to be estimated by the centralized quaternion Kalman filter (CQKF) given in Algorithm 2, where

$$\mathbf{H}_{col,n}^a = [\mathbf{H}_{1,n}^{aT}, \ldots, \mathbf{H}_{|\mathcal{N}|,n}^{aT}]^T$$

is the column block matrix of the augmented observation functions with $\mathbf{H}_{m,n}^a$ representing the observation function at node m and at time instant n, while $\mathbf{C}_{\boldsymbol{\omega}_{col,n}^a}$ is the augmented covariance matrix of the column vector of the combined augmented observation noises given by

$$\boldsymbol{\omega}_{col,n}^a = [\boldsymbol{\omega}_{1,n}^{aT}, \ldots, \boldsymbol{\omega}_{|\mathcal{N}|,n}^{aT}]^T$$

with $\boldsymbol{\omega}_{m,n}^a$ denoting the observation noise at node m and at time instant n.

Although the CQKF is optimal in the sense that it incorporates all the available information in the network, its operation requires inversions of large matrices and the transfer of all observation vectors to the central node, which burdens the central node with communication traffic and heavy computations. We next show that the operations of the CQKF can be replicated within a distributed framework by making the following assumptions:

1) the network is "*connected*", that is, there exists a path between any two given nodes in the network,
2) the observation noise at one node is uncorrelated with the observation noise at other nodes in the network.

Assuming that the observation noise at one node is uncorrelated with the observation noise at other nodes in the network leads to a block diagonal $\mathbf{C}_{\boldsymbol{\omega}_{col,n}^a}$ and therefore the *a posteriori*

Algorithm 2: Centralized Quaternion Kalman Filter (CQKF) [26].

Initialize with:

$$\hat{\mathbf{x}}^a_{0|0} = E[\mathbf{x}^a_0]$$

$$\hat{\mathbf{M}}^a_{0|0} = E\big[(\mathbf{x}^a_0 - E[\mathbf{x}^a_0])(\mathbf{x}^a_0 - E[\mathbf{x}^a_0])^H\big]$$

Model update:

$$\hat{\mathbf{x}}^a_{n|n-1} = \mathbf{A}^a_n \hat{\mathbf{x}}^a_{n-1|n-1}$$

$$\hat{\mathbf{M}}^a_{n|n-1} = \mathbf{A}^a_n \hat{\mathbf{M}}^a_{n-1|n-1} \mathbf{A}^{aH}_n + \mathbf{C}_{\boldsymbol{\nu}^a_n}$$

Measurement update:

$$\hat{\mathbf{M}}^{a^{-1}}_{n|n} = \hat{\mathbf{M}}^{a^{-1}}_{n|n-1} + \mathbf{H}^{aH}_{col,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{col,n}} \mathbf{H}^a_{col,n}$$

$$\mathbf{G}_n = \hat{\mathbf{M}}^a_{n|n} \mathbf{H}^{aH}_{col,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{col,n}}$$

$$\hat{\mathbf{x}}^a_{n|n} = \hat{\mathbf{x}}^a_{n|n-1} + \mathbf{G}_n \left(\mathbf{y}^a_{col,n} - \mathbf{H}^a_{col,n} \hat{\mathbf{x}}^a_{n|n-1}\right)$$

estimate of the augmented state vector can be expressed as

$$\hat{\mathbf{x}}^a_{n|n} = \hat{\mathbf{x}}^a_{n|n-1} + \sum_{\forall l \in \mathcal{N}} \hat{\mathbf{M}}^a_{n|n} \mathbf{H}^{aH}_{l,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{l,n}} \left(\mathbf{y}^a_{l,n} - \mathbf{H}^a_{l,n} \hat{\mathbf{x}}^a_{n|n-1}\right). \quad (4)$$

The key in moving from a centralized implementation to a distributed one is to formulate the *a posteriori* augmented state vector estimate, $\hat{\mathbf{x}}^a_{n|n}$, in the from of the network average of the local state vector estimates, $\boldsymbol{\phi}^a_{l,n}$, that is

$$\hat{\mathbf{x}}^a_{n|n} = \frac{1}{|\mathcal{N}|} \sum_{\forall l \in \mathcal{N}} \boldsymbol{\phi}^a_{l,n} \quad (5)$$

with $\boldsymbol{\phi}^a_{l,n}$ denoting the local estimate of the augmented state vector at node l at time instant n, which is given by

$$\boldsymbol{\phi}^a_{l,n} = \hat{\mathbf{x}}^a_{n|n-1} + |\mathcal{N}| \hat{\mathbf{M}}^a_{n|n} \mathbf{H}^{aH}_{l,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{l,n}} \left(\mathbf{y}^a_{l,n} - \mathbf{H}^a_{l,n} \hat{\mathbf{x}}^a_{n|n-1}\right) \quad (6)$$

where the Kalman filter update is scaled by a factor of $|\mathcal{N}|$ to preserve the equivalence of the centralized implementation in (4) and the distributed implementation in (5), (6). Furthermore, assuming uncorrelated observation noise throughout the network, from Algorithm 2, we have

$$\hat{\mathbf{M}}^{a^{-1}}_{n|n} = \hat{\mathbf{M}}^{a^{-1}}_{n|n-1} + \sum_{\forall l \in \mathcal{N}} \mathbf{H}^{aH}_{l,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}_{l,n}} \mathbf{H}^a_{l,n}. \quad (7)$$

Now, substituting (7) into (6) yields

$$\boldsymbol{\phi}^a_{l,n} = \hat{\mathbf{x}}^a_{n|n-1} + \mathbf{G}_{l,n} \left(\mathbf{y}^a_{l,n} - \mathbf{H}^a_{l,n} \hat{\mathbf{x}}^a_{n|n-1}\right) \quad (8)$$

where $\mathbf{G}_{l,n}$ is given by

$$\mathbf{G}_{l,n} = |\mathcal{N}| \left(\hat{\mathbf{M}}^{a^{-1}}_{n|n-1} + \sum_{\forall m \in \mathcal{N}} \mathbf{H}^{aH}_{m,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{m,n}} \mathbf{H}^a_{m,n}\right)^{-1} \mathbf{H}^{aH}_{l,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{l,n}}. \quad (9)$$

Making the assumption that the network is *connected*, allows $\sum_{\forall m \in \mathcal{N}} \mathbf{H}^{aH}_{m,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{m,n}} \mathbf{H}^a_{m,n}$ to be obtained through a diffusion of the local parameters $\mathbf{H}^{aH}_{l,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{l,n}} \mathbf{H}^a_{l,n}$. Thus, $\hat{\mathbf{M}}^a_{n|n}$ in the formulation in (7) and $\mathbf{G}_{l,n}$ in the formulation in (9) can be obtained at each node in the network in a distributed fashion, which in turn permits $\boldsymbol{\phi}^a_{l,n}$ in the formulation in (8) to be calculated by the individual nodes of the network. In addition, assuming a *connected* network permits $\hat{\mathbf{x}}^a_{n|n}$ to be obtained in a distributed manner by averaging local estimates, $\boldsymbol{\phi}^a_{l,n}$. Therefore, the operations of the CQKF can be mirrored in a distributed fashion through diffusion of the local parameters $\mathbf{H}^{aH}_{l,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{l,n}} \mathbf{H}^a_{l,n}$ and the averaging of local estimates $\boldsymbol{\phi}^a_{l,n}$. The operations of such a distributed quaternion Kalman filter (DQKF) are summarized in Algorithm 3, where $\mathcal{N}_l$ denotes the set of nodes in the neighborhood of node l.

Remark 2: The DQKF implemented at a node is optimal in the sense that it operates akin to a centralized Kalman filter combining all the information available to the nodes in its neighborhood.

Remark 3: In essence, we have shown that locally optimal Kalman filters can be implemented in a distributed fashion by sharing only the local estimates $\boldsymbol{\phi}^a_{l,n}$ and the parameters $\mathbf{H}^{aH}_{l,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{l,n}} \mathbf{H}^a_{l,n}$, whereas conventional distributed Kalman filtering techniques (see [2], [9]) require the sharing of additional information on local measurements, observation functions, and noise covariance matrices. In addition, in contrast to conventional distributed Kalman filtering techniques no extra mixing coefficients are required for averaging local estimates, $\boldsymbol{\phi}^a_{l,n}$ and the effect of diffusing local estimates, obtained through different observation functions and with different noise characteristics, on the *a posteriori* estimate of the error augmented covariance matrix, $\hat{\mathbf{M}}^a_{n|n}$, is taken into account.

IV. Performance Analysis

In order to analyze the mean and mean square performance of the developed algorithm, the error of the augmented state vector estimates is first expressed in a recursive manner. The difference between the true augmented state vector and the local estimate at node l and at time instant n is given by $\boldsymbol{\epsilon}^a_{l,n} = \mathbf{x}^a_n - \boldsymbol{\phi}^a_{l,n}$ which can be alternatively expressed as

$$\boldsymbol{\epsilon}^a_{l,n} = \mathbf{x}^a_n - \hat{\mathbf{x}}^a_{l,n|n-1} - \mathbf{G}_{l,n} \left(\mathbf{y}^a_{l,n} - \mathbf{H}^a_{l,n} \hat{\mathbf{x}}^a_{l,n|n-1}\right)$$

where upon replacing $\mathbf{y}^a_{l,n} = \mathbf{H}^a_{l,n} \mathbf{x}^a_n + \boldsymbol{\omega}^a_{l,n}$ and $\boldsymbol{\epsilon}^a_{l,n|n-1} = \mathbf{x}^a_n - \hat{\mathbf{x}}^a_{l,n|n-1}$ we have

$$\boldsymbol{\epsilon}^a_{l,n} = \left(\mathbf{I} - \mathbf{G}_{l,n} \mathbf{H}^a_{l,n}\right) \boldsymbol{\epsilon}^a_{l,n|n-1} - \mathbf{G}_{l,n} \boldsymbol{\omega}^a_{l,n}. \quad (10)$$

Algorithm 3: Distributed Quaternion Kalman Filter (DQKF).

For node $l = \{1, \cdots, |\mathcal{N}|\}$:

Initialize with:

$$\hat{\mathbf{x}}^a_{l,0|0} = E[\mathbf{x}^a_0]$$

$$\hat{\mathbf{M}}^a_{l,0|0} = E\big[(\mathbf{x}^a_0 - E[\mathbf{x}^a_0])(\mathbf{x}^a_0 - E[\mathbf{x}^a_0])^H\big]$$

Model update:

$$\hat{\mathbf{x}}^a_{l,n|n-1} = \mathbf{A}^a_n \hat{\mathbf{x}}^a_{l,n-1|n-1}$$

$$\hat{\mathbf{M}}^a_{l,n|n-1} = \mathbf{A}^a_n \hat{\mathbf{M}}^a_{l,n-1|n-1} \mathbf{A}^{aH}_n + \mathbf{C}_{\boldsymbol{\nu}^a_n}$$

Measurement update:

$$\hat{\mathbf{M}}^{a^{-1}}_{l,n|n} = \hat{\mathbf{M}}^{a^{-1}}_{l,n|n-1} + \sum_{\forall m \in \mathcal{N}_l} \Big(\mathbf{H}^{aH}_{m,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{m,n}} \mathbf{H}^a_{m,n} \Big)$$

$$\mathbf{G}_{l,n} = |\mathcal{N}_l| \hat{\mathbf{M}}^a_{l,n|n} \mathbf{H}^{aH}_{l,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{l,n}}$$

$$\boldsymbol{\phi}^a_{l,n} = \hat{\mathbf{x}}^a_{l,n|n-1} + \mathbf{G}_{l,n} \big(\mathbf{y}^a_{l,n} - \mathbf{H}^a_{l,n} \hat{\mathbf{x}}^a_{l,n|n-1} \big)$$

Information sharing:

1) Share $\boldsymbol{\phi}^a_{l,n}$ with neighboring nodes.
2) Share $\mathbf{H}^{aH}_{l,n} \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{l,n}} \mathbf{H}^a_{l,n}$ with neighboring nodes, only if it has changed compared to the previous time instant.

Averaging:

$$\hat{\mathbf{x}}^a_{l,n|n} = \frac{1}{|\mathcal{N}_l|} \sum_{\forall m \in \mathcal{N}_l} \boldsymbol{\phi}^a_{m,n}$$

Furthermore, substituting $\boldsymbol{\epsilon}^a_{l,n|n-1} = \mathbf{A}^a_n \boldsymbol{\epsilon}^a_{l,n-1|n-1} + \boldsymbol{\nu}^a_n$ into (10) gives

$$\begin{aligned} \boldsymbol{\epsilon}^a_{l,n} =\ & \big(\mathbf{I} - \mathbf{G}_{l,n} \mathbf{H}^a_{l,n}\big) \mathbf{A}^a_n \boldsymbol{\epsilon}^a_{l,n-1|n-1} \\ & + \big(\mathbf{I} - \mathbf{G}_{l,n} \mathbf{H}^a_{l,n}\big) \boldsymbol{\nu}^a_n - \mathbf{G}_{l,n} \boldsymbol{\omega}^a_{l,n}. \end{aligned} \tag{11}$$

Now, consider the difference between the true augmented state vector and its estimate obtained at node l, given by

$$\begin{aligned} \boldsymbol{\epsilon}^a_{l,n|n} &= \mathbf{x}^a_n - \hat{\mathbf{x}}^a_{l,n|n} \\ &= \mathbf{x}^a_n - \frac{1}{|\mathcal{N}_l|} \sum_{\forall m \in \mathcal{N}_l} \boldsymbol{\phi}^a_{m,n} = \frac{1}{|\mathcal{N}_l|} \sum_{\forall m \in \mathcal{N}_l} \boldsymbol{\epsilon}^a_{m,n} \end{aligned} \tag{12}$$

where replacing (11) into (12) gives a recursive expression for the augmented state vector estimation error as

$$\begin{aligned} \boldsymbol{\epsilon}^a_{l,n|n} =\ & \frac{1}{|\mathcal{N}_l|} \sum_{\forall m \in \mathcal{N}_l} \big(\mathbf{I} - \mathbf{G}_{m,n} \mathbf{H}^a_{m,n}\big) \mathbf{A}^a_n \boldsymbol{\epsilon}^a_{m,n-1|n-1} \\ & + \frac{1}{|\mathcal{N}_l|} \sum_{\forall m \in \mathcal{N}_l} \big(\mathbf{I} - \mathbf{G}_{m,n} \mathbf{H}^a_{m,n}\big) \boldsymbol{\nu}^a_n \\ & - \frac{1}{|\mathcal{N}_l|} \sum_{\forall m \in \mathcal{N}_l} \mathbf{G}_{m,n} \boldsymbol{\omega}^a_{m,n}. \end{aligned} \tag{13}$$

From Algorithm 3, we can now substitute

$$\mathbf{G}^a_{m,n} \mathbf{H}^a_{m,n} = \hat{\mathbf{M}}^a_{m,n|n} \underbrace{\mathbf{H}^{aH}_{m,n} |\mathcal{N}_m| \mathbf{C}^{-1}_{\boldsymbol{\omega}^a_{m,n}} \mathbf{H}^a_{m,n}}_{\mathbf{P}_{m,n}}$$

into (13) to yield

$$\begin{aligned} \boldsymbol{\epsilon}^a_{l,n|n} =\ & \frac{1}{|\mathcal{N}_l|} \sum_{\forall m \in \mathcal{N}_l} \Big(\mathbf{I} - \hat{\mathbf{M}}^a_{m,n|n} \mathbf{P}_{m,n}\Big) \mathbf{A}^a_n \boldsymbol{\epsilon}^a_{m,n-1|n-1} \\ & + \frac{1}{|\mathcal{N}_l|} \sum_{\forall m \in \mathcal{N}_l} \Big(\mathbf{I} - \hat{\mathbf{M}}^a_{m,n|n} \mathbf{P}_{m,n}\Big) \boldsymbol{\nu}^a_n \\ & - \frac{1}{|\mathcal{N}_l|} \sum_{\forall m \in \mathcal{N}_l} \mathbf{G}_{m,n} \boldsymbol{\omega}^a_{m,n}. \end{aligned} \tag{14}$$

A. Mean Error Behavior

Taking the statistical expectation of (14) and noting that $\boldsymbol{\nu}^a_n$ and $\boldsymbol{\omega}^a_{m,n}$ are zero-mean results in

$$E[\boldsymbol{\epsilon}^a_{l,n|n}] = \frac{1}{|\mathcal{N}_l|} \sum_{\forall m \in \mathcal{N}_l} \Big(\mathbf{I} - \hat{\mathbf{M}}^a_{m,n|n} \mathbf{P}_{m,n}\Big) \mathbf{A}^a_n E[\boldsymbol{\epsilon}^a_{m,n-1|n-1}]. \tag{15}$$

Therefore, given that $\forall m \in \mathcal{N} : \hat{\mathbf{x}}^a_{m,0|0} = E[\mathbf{x}^a_0]$, the expression in (15) indicates that the algorithm operates in an unbiased fashion.

B. Local Mean Square Error Behavior

Given the error of the augmented state vector estimates in the formulation in (14), the augmented error covariance matrix of the augmented state vector estimates at node l and time instant n can be expressed as

$$\begin{aligned} \boldsymbol{\Sigma}^a_{l,n} &= E\left[\boldsymbol{\epsilon}^a_{l,n|n} \boldsymbol{\epsilon}^{aH}_{l,n|n}\right] \\ &= \mathbf{S}_{l,n} \mathcal{E}_{n-1} \mathbf{S}^H_{l,n} + \mathbf{R}_{l,n} \mathcal{V}_n \mathbf{R}^H_{l,n} + \mathbf{Q}_{l,n} \mathcal{W}_n \mathbf{Q}^H_{l,n} \end{aligned} \tag{16}$$

where the expressions

$$\begin{aligned} \mathcal{E}_n &= E\left[\left[\boldsymbol{\epsilon}^{aT}_{1,n|n}, \ldots, \boldsymbol{\epsilon}^{aT}_{|\mathcal{N}|,n|n}\right]^T \left[\boldsymbol{\epsilon}^{aT}_{1,n|n}, \ldots, \boldsymbol{\epsilon}^{aT}_{|\mathcal{N}|,n|n}\right]^*\right] \\ \mathcal{W}_n &= E\left[\left[\boldsymbol{\omega}^{aT}_{1,n}, \ldots, \boldsymbol{\omega}^{aT}_{|\mathcal{N}|,n}\right]^T \left[\boldsymbol{\omega}^{aT}_{1,n}, \ldots, \boldsymbol{\omega}^{aT}_{|\mathcal{N}|,n}\right]^*\right] \\ \mathcal{V}_n &= \text{block}(\mathbf{C}_{\boldsymbol{\nu}^a_n}) = \begin{bmatrix} \mathbf{C}_{\boldsymbol{\nu}^a_n} & \cdots & \mathbf{C}_{\boldsymbol{\nu}^a_n} \\ \vdots & \ddots & \vdots \\ \mathbf{C}_{\boldsymbol{\nu}^a_n} & \cdots & \mathbf{C}_{\boldsymbol{\nu}^a_n} \end{bmatrix} \end{aligned} \tag{17}$$

represent respectively the state estimation error cross-covariance between all nodes in the network, the observation noise cross-covariances between all nodes in the network, and

a block matrix with all its elements equal to $\mathbf{C}_{\boldsymbol{\nu}_n^a}$, while

$$\mathbf{S}_{l,n} = \frac{1}{|\mathcal{N}_l|}\begin{bmatrix} \alpha_{l,1}\mathbf{A}_n^{aH}\left(\mathbf{I}-\hat{\mathbf{M}}_{1,n|n}^a\mathbf{P}_{1,n}\right)^H \\ \alpha_{l,2}\mathbf{A}_n^{aH}\left(\mathbf{I}-\hat{\mathbf{M}}_{2,n|n}^a\mathbf{P}_{2,n}\right)^H \\ \vdots \\ \alpha_{l,|\mathcal{N}|}\mathbf{A}_n^{aH}\left(\mathbf{I}-\hat{\mathbf{M}}_{|\mathcal{N}|,n|n}^a\mathbf{P}_{|\mathcal{N}|,n}\right)^H \end{bmatrix}^H$$

$$\mathbf{R}_{l,n} = \frac{1}{|\mathcal{N}_l|}\begin{bmatrix} \alpha_{l,1}\left(\mathbf{I}-\hat{\mathbf{M}}_{1,n|n}^a\mathbf{P}_{1,n}\right)^H \\ \alpha_{l,2}\left(\mathbf{I}-\hat{\mathbf{M}}_{2,n|n}^a\mathbf{P}_{2,n}\right)^H \\ \vdots \\ \alpha_{l,|\mathcal{N}|}\left(\mathbf{I}-\hat{\mathbf{M}}_{|\mathcal{N}|,n|n}^a\mathbf{P}_{|\mathcal{N}|,n}\right)^H \end{bmatrix}^H \tag{18}$$

$$\mathbf{Q}_{l,n} = \frac{1}{|\mathcal{N}_l|}\left[\alpha_{l,1}\mathbf{G}_{1,n}, \alpha_{1,2}\mathbf{G}_{2,n}, \cdots, \alpha_{1,|\mathcal{N}|}\mathbf{G}_{|\mathcal{N}|,n}\right]$$

with

$$\alpha_{l,m} = \begin{cases} 1, & \text{if } m \in \mathcal{N}_l \\ 0, & \text{otherwise.} \end{cases}$$

We shall now use the following standard assumptions in steady-state Kalman filtering analysis [40]:

1) the state evolution function and observation functions for all nodes in the network become time invariant, i.e.
$$\forall l \in \mathcal{N}: \begin{cases} \lim_{n\to\infty} \mathbf{A}_n^a = \mathbf{A}^a \\ \lim_{n\to\infty} \mathbf{H}_{l,n}^a = \mathbf{H}_l^a \end{cases}$$
2) the state evolution and observation noises are stationary, that is, $\mathbf{C}_{\boldsymbol{\nu}_n^a} \to \mathbf{C}_{\boldsymbol{\nu}^a}$ and $\mathcal{W}_n \to \mathcal{W}$, which in turn implies that $\mathcal{V}_n \to \mathcal{V}$ and $\forall l \in \mathcal{N}: \mathbf{C}_{\boldsymbol{\omega}_{l,n}^a} \to \mathbf{C}_{\boldsymbol{\omega}_l^a}$.
3) the matrix pairs $\forall l \in \mathcal{N}: \{\mathbf{A}^a, \mathbf{H}_{l,n}^a\}$ are observable and the matrix pair $\{\mathbf{A}^a, \mathbf{C}_{\boldsymbol{\nu}^a}^{\frac{1}{2}}\}$ is controllable.

Then, it follows that for all nodes in the network, $\hat{\mathbf{M}}_{l,n|n}^a$ becomes time invariant. Moreover, from the expressions in (18), a time invariant $\hat{\mathbf{M}}_{l,n|n}^a$ would result in the matrices $\{\mathbf{S}_{l,n}, \mathbf{R}_{l,n}, \mathbf{Q}_{l,n}\}$ also becoming time invariant, which can be summarized as

$$\text{Given } \lim_{n\to\infty} \hat{\mathbf{M}}_{l,n|n}^a = \hat{\mathbf{M}}_l^a \Rightarrow \forall l \in \mathcal{N}: \begin{matrix} \lim_{n\to\infty} \mathbf{S}_{l,n} = \mathbf{S}_l \\ \lim_{n\to\infty} \mathbf{R}_{l,n} = \mathbf{R}_l \\ \lim_{n\to\infty} \mathbf{Q}_{l,n} = \mathbf{Q}_l \end{matrix}$$

and therefore $\boldsymbol{\Sigma}_{l,n}^a$ converges.

Remark 4: From Algorithm 3 and the expression in (16), notice that the correlation between the observation noise at different nodes in the network does not have an effect on $\mathbf{S}_{l,n}$, $\mathbf{R}_{l,n}$, and $\mathbf{Q}_{l,n}$. In addition, $\text{Tr}(\boldsymbol{\Sigma}_{l,n}^a)$ is linearly dependent on $\text{Tr}(\mathbf{Q}_{l,n}\mathcal{W}_n\mathbf{Q}_{l,n}^H)$. Hence, for a constant value of $\text{Tr}(\mathcal{W}_n)$, we have that $\text{Tr}(\mathbf{Q}_{l,n}\mathcal{W}_n\mathbf{Q}_{l,n}^H)$ and therefore $\text{Tr}(\boldsymbol{\Sigma}_{l,n}^a)$ are minimized (*cf.* maximized) when the observation noises at different nodes are uncorrelated (*cf.* fully correlated).

Remark 5: Note that since $\{\mathbf{S}_{l,n}, \mathbf{R}_{l,n}, \mathbf{Q}_{l,n}\}$ given in (18) are linearly dependent on $|\mathcal{N}_l|$, which is referred to as the connection degree of the node. Therefore, the term $\boldsymbol{\Sigma}_{l,n}^a$ in (16) will also be dependent on the connection degree of the node.

C. Global Mean Square Error Behavior

From the expression in (14), the estimation error cross-covariance between all nodes of the network, $\mathcal{E}_n$, can be given in a recursive formulation as

$$\mathcal{E}_n = \mathcal{S}_n\mathcal{E}_{n-1}\mathcal{S}_n^H + \mathcal{R}_n\mathcal{V}_n\mathcal{R}_n^H + \mathcal{Q}_n\mathcal{W}_n\mathcal{Q}_n^H \tag{19}$$

where $\mathcal{E}_n$, $\mathcal{V}_n$, and $\mathcal{W}_n$ are given in (17), while

$$\mathcal{S}_n = \begin{bmatrix} \mathbf{S}_{1,n} \\ \vdots \\ \mathbf{S}_{|\mathcal{N}|,n} \end{bmatrix}, \mathcal{R}_n = \begin{bmatrix} \mathbf{R}_{1,n} \\ \vdots \\ \mathbf{R}_{|\mathcal{N}|,n} \end{bmatrix}, \text{and } \mathcal{Q}_n = \begin{bmatrix} \mathbf{Q}_{1,n} \\ \vdots \\ \mathbf{Q}_{|\mathcal{N}|,n} \end{bmatrix}.$$

Then, if convergence conditions in Section IV-B are satisfied, i.e. $\mathcal{V}_n \to \mathcal{V}$ and $\mathcal{W}_n \to \mathcal{W}$, as a result of local convergence, matrices $\{\mathcal{S}_n, \mathcal{R}_n, \mathcal{Q}_n\}$ become time invariant, that is

$$\lim_{n\to\infty} \mathcal{S}_n = \mathcal{S}, \ \lim_{n\to\infty} \mathcal{R}_n = \mathcal{R}, \text{ and } \lim_{n\to\infty} \mathcal{Q}_n = \mathcal{Q}$$

and $\mathcal{E}_n$ in (19) converges, that is, $\mathcal{E}_n \to \mathcal{E}$ as $n \to \infty$. Therefore, the expression in (19) simplifies to a quaternion-valued discrete time Lyanpunov equation given by

$$\mathcal{E} = \mathcal{S}\mathcal{E}\mathcal{S}^H + \mathcal{R}\mathcal{V}\mathcal{R}^H + \mathcal{Q}\mathcal{W}\mathcal{Q}^H. \tag{20}$$

Invoking the duality between $\mathbb{R}$ and $\mathbb{H}$ established using the expressions in (1) and through decomposing the quaternion-valued matrices in (20) into their real-valued components, the closed form solution to the equation in (20) can be obtained as

$$\text{Vec}(\mathcal{E}^{\mathbb{HR}}) = \left(\mathcal{I} - \mathcal{S}^{\mathbb{HR}} \otimes \mathcal{S}^{\mathbb{HR}}\right)^{-1} \text{Vec}\left(\mathcal{A}^{\mathbb{HR}}\right) \tag{21}$$

where $\mathcal{A} = \mathcal{R}\mathcal{V}\mathcal{R}^H + \mathcal{Q}\mathcal{W}\mathcal{Q}^H$ and $\mathcal{I}$ is an identity matrix with the same number of rows as $\mathcal{S}^{\mathbb{HR}} \otimes \mathcal{S}^{\mathbb{HR}}$, while

$$\mathcal{E}^{\mathbb{HR}} = \begin{bmatrix} \mathcal{E}_r & -\mathcal{E}_i & -\mathcal{E}_j & -\mathcal{E}_k \\ \mathcal{E}_i & \mathcal{E}_r & -\mathcal{E}_k & \mathcal{E}_j \\ \mathcal{E}_j & \mathcal{E}_k & \mathcal{E}_r & -\mathcal{E}_i \\ \mathcal{E}_k & -\mathcal{E}_j & \mathcal{E}_i & \mathcal{E}_r \end{bmatrix}$$

with $\mathcal{S}^{\mathbb{HR}}$ and $\mathcal{A}^{\mathbb{HR}}$ defined analogously.

D. Extension to Multi-Task Networks

Note that in Algorithm 3 it is assumed that all the nodes in the network are estimating the same augmented state vector sequence; however, in many applications this assumption may not hold true and therefore it becomes necessary for a node to identify other nodes in its neighborhood that are estimating the same augmented state vector sequence. Let $\mathbf{x}_{l,n}^a$ and $\mathbf{x}_{m,n}^a$ denote respectively the augmented state vectors of nodes l and m at time instant n. Considering that $\boldsymbol{\epsilon}_{l,n|n-1}^a = \mathbf{x}_{l,n}^a - \hat{\mathbf{x}}_{l,n|n-1}^a$; from Algorithm 3, $\boldsymbol{\epsilon}_{l,n|n-1}^a$ is a zero-mean quaternion-valued Gaussian random vector with the augmented covariance matrix $\hat{\mathbf{M}}_{l,n|n-1}^a$. Therefore, from (13) and considering that $\boldsymbol{\omega}_{l,n}^a$ is a quaternion-valued zero-mean Gaussian random vector, $\boldsymbol{\epsilon}_{l,n}^a =$

$\mathbf{x}^a_{l,n} - \boldsymbol{\phi}^a_{l,n}$ will also be a zero-mean quaternion-valued random vector with the augmented covariance matrix

$$\mathbf{C}_{\boldsymbol{\epsilon}^a_{l,n}} = \left(\mathbf{I} - \mathbf{G}_{l,n}\mathbf{H}^a_{l,n}\right)\hat{\mathbf{M}}^a_{l,n|n-1}\left(\mathbf{I} - \mathbf{G}_{l,n}\mathbf{H}^a_{l,n}\right)^H + \mathbf{G}_{l,n}\mathbf{C}_{\boldsymbol{\omega}^a_{l,n}}\mathbf{G}^H_{l,n}. \quad (22)$$

Now, consider a measure of difference between the observation at node m and its predicted value given the local estimate at node l, defined as

$$\begin{aligned}\mathbf{r}^a_{(l,m)} &= \mathbf{H}^a_{m,n}\boldsymbol{\phi}^a_{l,n} - \mathbf{y}^a_{m,n} = \mathbf{H}^a_{m,n}\left(\boldsymbol{\phi}^a_{l,n} - \mathbf{x}^a_{m,n}\right) - \boldsymbol{\omega}^a_{l,n} \\ &= \mathbf{H}^a_{m,n}\Delta\mathbf{x}^a_{(l,m)_n} - \mathbf{H}^a_{m,n}\boldsymbol{\epsilon}^a_{l,n} - \boldsymbol{\omega}^a_{l,n}\end{aligned}$$

where $\Delta x^a_{(l,m)_n}$ denotes the difference between the augmented state vectors at nodes l and m at time instant n. Note that $\mathbf{r}^a_{(l,m)}$ is a quaternion-valued Gaussian random vector with the augmented covariance matrix

$$\mathbf{C}_{\mathbf{r}^a_{(l,m)}} = \mathbf{H}^a_{m,n}\mathbf{C}_{\boldsymbol{\epsilon}^a_{l,n}}\mathbf{H}^{aH}_{m,n} + \mathbf{C}_{\boldsymbol{\omega}^a_{l,n}} \quad (23)$$

and mean vector $\mathbf{H}^a_{m,n}\Delta\mathbf{x}^a_{(l,m)_n}$, where $\mathbf{C}_{\boldsymbol{\epsilon}^a_{l,n}}$ is given in (22).

In the case when the nodes l and m are estimating the same augmented state vector sequence; $\Delta\mathbf{x}^a_{(l,m)_n} = 0$ and hence $\mathbf{H}^a_{m,n}\Delta\mathbf{x}^a_{(l,m)_n} = 0$. Therefore, the Mahalanobis distance $d = \mathbf{r}^{aH}_{(l,m)}\mathbf{C}^{-1}_{\mathbf{r}^a_{(l,m)}}\mathbf{r}^a_{(l,m)}$ can be used as a confidence measure to indicate whether $\mathbf{r}^a_{(l,m)}$ is an outlier (*cf.* not an outlier) for a zero-mean quaternion-valued distribution with the augmented covariance matrix $\mathbf{C}_{\mathbf{r}^a_{(l,m)}}$ indicating that the local estimate at node l, $\boldsymbol{\phi}^a_{l,n}$, offers an invalid (*cf.* valid) update for the augmented state vector estimates at node m given the measurement $\mathbf{y}^a_{m,n}$.

Remark 6: Note that if $\Delta\mathbf{x}^a_{(l,m)_n} = 0$; then, d is a *chi-square* random variable whereas for the case where $\Delta\mathbf{x}^a_{(l,m)_n} \neq 0$ the confidence measure, d, is a non-central *chi-square* random variable. This allows the probabilities of miss-detection and false alarm to be established using numerical methods for given values of $\Delta\mathbf{x}^a_{(l,m)_n}$ [41], [42]. Since, in most applications prior knowledge of $\Delta\mathbf{x}^a_{(l,m)_n}$ is not available, in the solution devised here, the threshold is set to be the 90% probability mass line of a zero-mean quaternion-valued Gaussian random variable with the augmented covariance matrix given in (23) essentially assuming that the threshold is satisfied (*cf.* not satisfied) due to the fact that $\Delta\mathbf{x}^a_{(l,m)_n} = 0$ (*cf.* $\Delta\mathbf{x}^a_{(l,m)_n} \neq 0$).

V. Applications

The newly developed DQKF was applied for frequency estimation in three-phase power distribution networks and for collaborative target tracking. In the simulations, the network of 20 nodes shown in Figure 1 was used, where the estimates of the system frequency or target location from the node denoted by the red circle were used for illustrating the performance of the DQKF.

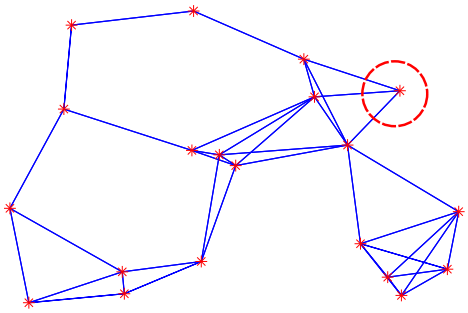

Fig. 1. The network of 20 nodes used in simulations, where the nodes are marked by red "*" and the connections are shown with blue lines. The node for which the estimates of the system frequency or target location are illustrated in later sections is marked by a red circle.

A. Frequency Estimation in Smart Grids

Consider the instantaneous voltages of each phase in a three-phase power system, given by [43]

$$\begin{aligned}v_{a,n} &= V_{a,n}\sin\left(2\pi f\Delta T n + \theta_{a,n}\right) \\ v_{b,n} &= V_{b,n}\sin\left(2\pi f\Delta T n + \theta_{b,n} + \frac{2\pi}{3}\right) \\ v_{c,n} &= V_{c,n}\sin\left(2\pi f\Delta T n + \theta_{c,n} + \frac{4\pi}{3}\right)\end{aligned} \quad (24)$$

where $V_{a,n}$, $V_{b,n}$, and $V_{c,n}$ are the instantaneous amplitudes, $\theta_{a,n}$, $\theta_{b,n}$, and $\theta_{c,n}$ represent the instantaneous phase shifts, and ΔT is the sampling interval, while f denotes the system frequency. The three-phase system is referred to as balanced if $V_{a,n} = V_{b,n} = V_{c,n}$ and $\theta_{a,n} = \theta_{b,n} = \theta_{c,n}$. The power grid is designed to operate optimally at a nominal frequency and in a balanced fashion. Large deviations from the nominal frequency and unbalanced operating conditions can adversely affect the performance of different components of the power grid, such as compensators and loads [44], [45], resulting in harmful operating conditions that can propagate throughout the network. In addition, real-time frequency tracking reveals essential information about the dynamics of the power grid, such as power generation-consumption mismatch. Therefore, smart grid control and management applications require accurate estimates of the power signal frequency in order to ensure nominal operating conditions [18], [27].

For more than 50 years the standard approach for the analysis of three-phase power systems has been to apply the Clarke transform, given by [43]

$$\begin{bmatrix} v_{0,n} \\ v_{\alpha,n} \\ v_{\beta,n} \end{bmatrix} = \sqrt{\frac{2}{3}}\begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix}\begin{bmatrix} v_{a,n} \\ v_{b,n} \\ v_{c,n} \end{bmatrix} \quad (25)$$

for mapping the three-phase voltages onto a new domain where they are represented by the complex number $v_n = v_{\alpha,n} + iv_{\beta,n}$. Note that due to the lack of dimensionality of complex numbers, $v_{0,n}$ has to be ignored in practical applications, which compromises complex-valued analysis techniques when it comes to dealing with three-phase systems as they cannot fully incorporate the available information.

Quaternions were used in [27] for modeling three-phase power signals in order to provide a framework for incorporating all the available information, where the three-phase voltages in (24) were combined together to generate the quaternion signal

$$q_n = iv_{a,n} + jv_{b,n} + kv_{c,n}. \tag{26}$$

A simple mathematical manipulation on (26) yields

$$q_n = \Lambda_{I,n}\cos(2\pi\Delta T n) + \Lambda_{Q,n}\sin(2\pi\Delta T n) \tag{27}$$

where $\Lambda_{I,n}$ and $\Lambda_{Q,n}$ are given by

$$\Lambda_{I,n} = iV_{a,n}\sin(\theta_{a,n}) + jV_{b,n}\sin\left(\theta_{b,n} + \frac{2\pi}{3}\right) + kV_{c,n}\sin\left(\theta_{c,n} + \frac{4\pi}{3}\right) \tag{28}$$

$$\Lambda_{Q,n} = iV_{a,n}\cos(\theta_{a,n}) + jV_{b,n}\cos\left(\theta_{b,n} + \frac{4\pi}{3}\right) + kV_{c,n}\cos\left(\theta_{c,n} + \frac{4\pi}{3}\right).$$

Replacing the $\sin(\cdot)$ and $\cos(\cdot)$ functions with their polar representations, gives

$$q_n = \frac{\Lambda_{I,n}}{2}\left(e^{(\gamma 2\pi f\Delta T n)} + e^{-(\gamma 2\pi f\Delta T n)}\right) + \frac{\Lambda_{Q,n}}{2\gamma}\left(e^{(\gamma 2\pi f\Delta T n)} - e^{-(\gamma 2\pi f\Delta T n)}\right) \tag{29}$$

where $\gamma = \Im\left(\Lambda_{I,n}\Lambda_{Q,n}\right) / \left|\Im\left(\Lambda_{I,n}\Lambda_{Q,n}\right)\right|$. Upon rearranging the expression in (29), we have

$$q_n = \underbrace{\left(\frac{\Lambda_{I,n}}{2} + \frac{\Lambda_{Q,n}}{2\gamma}\right)e^{(\gamma 2\pi f\Delta T n)}}_{q_n^+} + \underbrace{\left(\frac{\Lambda_{I,n}}{2} - \frac{\Lambda_{Q,n}}{2\gamma}\right)e^{-(\gamma 2\pi f\Delta T n)}}_{q_n^-} \tag{30}$$

where q_n has been divided into the two counter-rotating signals q_n^+ and q_n^-, which can be expressed by the quaternion linear regressions

$$q_n^+ = q_{n-1}^+ e^{\gamma 2\pi f\Delta T} \text{ and } q_n^- = q_{n-1}^- e^{-\gamma 2\pi f\Delta T}. \tag{31}$$

Taking into account the quaternion linear regressions in (31), where the phase incrementing element of q_n^+ is the quaternion conjugate of the phase incrementing element of q_n^-, a state space model for q_n is proposed in Algorithm 4, where $\varphi_n = e^{\gamma 2\pi f\Delta T}$, $\boldsymbol{\nu}_n$ is the state evolution noise, and ω_n the observation noise. Note that in all simulations in this section the sampling interval was considered to be $\Delta T = 0.001$ s.

Remark 7: For a balanced three-phase system it can be shown that $\gamma = (i + j + k)/\sqrt{3}$ and $q_n^- = 0$ (*cf.* $q_n^+ = 0$) if the system is positive sequenced (*cf.* negative sequenced) allowing to detect the incidences when the three-phase system is operating under unbalanced conditions and take appropriate action for restoring balanced operating conditions. Since the output of the Clarke transform does not incorporate all the available information in the three-phase signal, the operating conditions of the power system is not retrievable in complex-valued frequency estimators based on the Clarke transform.

Algorithm 4: State space model used for frequency estimation.

State evolution function: $\begin{bmatrix}\varphi_n\\ q_n^+\\ q_n^-\end{bmatrix} = \begin{bmatrix}\varphi_{n-1}\\ q_{n-1}^+\varphi_{n-1}\\ q_{n-1}^-\varphi_{n-1}^*\end{bmatrix} + \boldsymbol{\nu}_n$

Observation function: $q_n = \begin{bmatrix}0 & 1 & 1\end{bmatrix}\begin{bmatrix}\varphi_n\\ q_n^+\\ q_n^-\end{bmatrix} + \omega_n$

Estimate of frequency: $\hat{f}_n = \frac{1}{2\pi\Delta T}\Im\left(\ln\left(\varphi_n\right)\right)$

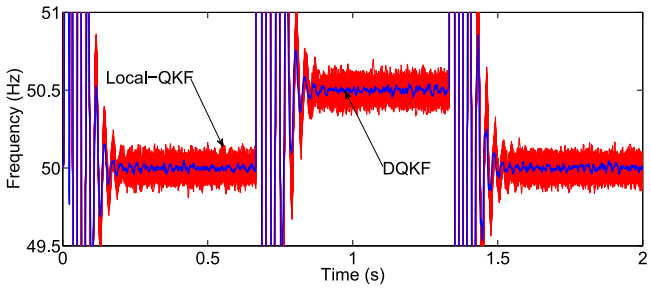

Fig. 2. Frequency estimation using the DQKF and the local-QKF. The estimate of the system frequency obtained by the DQKF is in solid blue line and the estimates obtained by the local-QKF are given in red.

In the first simulation, the three-phase system was considered to be initially operating at its nominal frequency of 50 Hz in a balanced fashion; then, the system suffered a fault resulting in unbalanced operating conditions characterized by an 80% drop in the amplitude of $v_{a,n}$ and 20 degree shifts in the phases of $v_{b,n}$ and $v_{c,n}$; furthermore, the frequency of the system experienced a step jump of 0.5 Hz. The fault lasted for a short duration and the system returned to its balanced operating condition and its nominal frequency. The estimates of the system frequency at the node indicated in the network shown in Figure 1, obtained through implementing a local-QKF and the newly developed DQKF are shown in Figure 2. Note that the estimates of the system frequency obtained through implementing the DQKF have significantly lower steady-state variance as compared to those obtained by the local-QKF.

In the second simulation, the three-phase system experiences a rise (*cf.* fall) in frequency due to a mismatch between power generation and consumption, while operating under the same unbalanced conditions characterized in the first simulation. In Figure 3, the estimates of the system frequency obtained at the node indicated in the network shown in Figure 1, through implementing a local-QKF are compared to those obtained through implementing the DQKF. Observe that the developed DQKF accurately tracked the system frequency and achieved a lower steady state variance as compared to the local-QKF due to cooperation between nodes in the network.

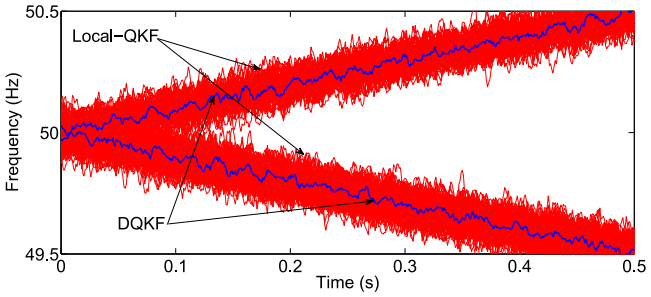

Fig. 3. Frequency estimation for an unbalanced three-phase system with changing frequency at the rate of 1 Hz/s. The estimate of the system frequency obtained using the DQKF is in solid blue line and the estimates obtained using the local-QKF are given in red.

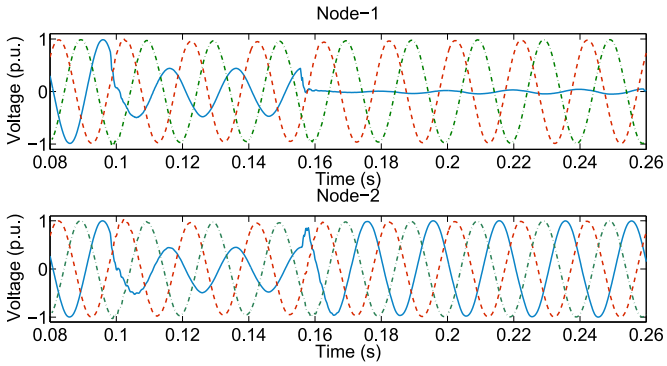

Fig. 4. Voltage recordings from two neighboring nodes in a real-world power distribution network.

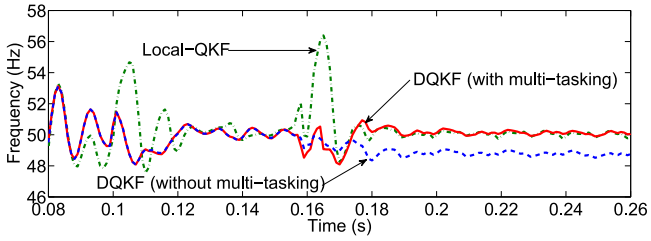

Fig. 5. Frequency estimation in a power distribution network using real-world data from two neighboring nodes.

In the third simulation, frequency estimation using real-world data recorded from two neighboring nodes in a power distribution network was considered. The recorded data are shown in Figure 4, where both nodes suffered a fault 0.1 second after the recording started. Although Node-2 recovered, Node-1 continued to operate in an unbalanced fashion. The estimates of the system frequency with and without the proposed multi-tasking technique are shown in Figure 5. Observe that the developed algorithm was able to detect that the nodes are operating under different circumstances and isolated their local estimators preventing bias in the estimated frequency.

In Figure 6, the mean square error (MSE) performance of the proposed quaternion frequency estimator, implemented using a local-QKF, the newly developed DQKF, and the CQKF is compared to that of its complex-valued counterparts that use the linear complex Kalman filter (LCKF) and widely-linear

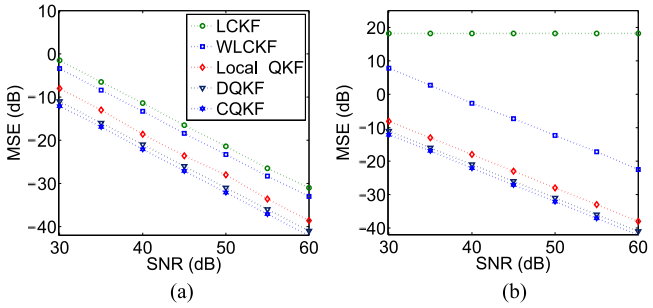

Fig. 6. Mean square error performance of various frequency estimation algorithms: a) balanced three-phase system, b) unbalanced three-phase system characterized by an 80% drop in the amplitude of $v_{a,n}$ and 20 degree shifts in the phases of $v_{b,n}$ and $v_{c,n}$.

complex Kalman filter (WLCKF) (see [46], [47]). Notice that the quaternion frequency estimator not only outperformed its linear and widely-linear complex-valued counterparts, but also the unbalanced operating conditions did not affect the performance of the quaternion frequency estimator, a desirable characteristic for frequency estimators in three-phase systems. Furthermore, employing the developed quaternion frequency estimator in its distributed form reduced the MSE by approximately 4dB. Implementing the quaternion frequency estimator through the CQKF only improved the steady-state variance performance by around 1dB as compared to implementing the quaternion frequency estimator through the DQKF.

B. Collaborative Target Tracking

We next considered the problem of tracking the location of a target in the three-dimensional space. To this end, consider the state vector [26]

$$\mathbf{x}_n = \begin{bmatrix} iL_{x_n} + jL_{y_n} + kL_{z_n} \\ i\dot{L}_{x_n} + j\dot{L}_{y_n} + k\dot{L}_{z_n} \end{bmatrix}$$

where $\{L_{x_n}, L_{y_n}, L_{y_n}\}$ and $\{\dot{L}_{x_n}, \dot{L}_{y_n}, \dot{L}_{z_n}\}$ denote the location and speed of the target along the X, Y, and Z axes. The state evolution function now becomes

$$\mathbf{x}_{n+1} = \underbrace{\begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix}}_{\mathbf{A}} \mathbf{x}_n + \underbrace{\begin{bmatrix} \frac{1}{2}(\Delta T)^2 \\ \Delta T \end{bmatrix}}_{\mathbf{B}} \eta_n \tag{32}$$

where ΔT denotes the sampling interval and $\eta_n = i\eta_{i,n} + j\eta_{j,n} + k\eta_{k,n}$ is a zero-mean quaternion-valued random variable used to model acceleration. The nodes in the network shown in Figure 1 are attempting to estimate the state vector through observations

$$\forall l \in \mathcal{N} : y_{l,n} = \underbrace{\begin{bmatrix} 0 & 1 \end{bmatrix}}_{\mathbf{H}} \mathbf{x}_n + \omega_{l,n} \tag{33}$$

which can be achieved by quaternion-valued Kalman filtering [26], where the state involution and observation equations

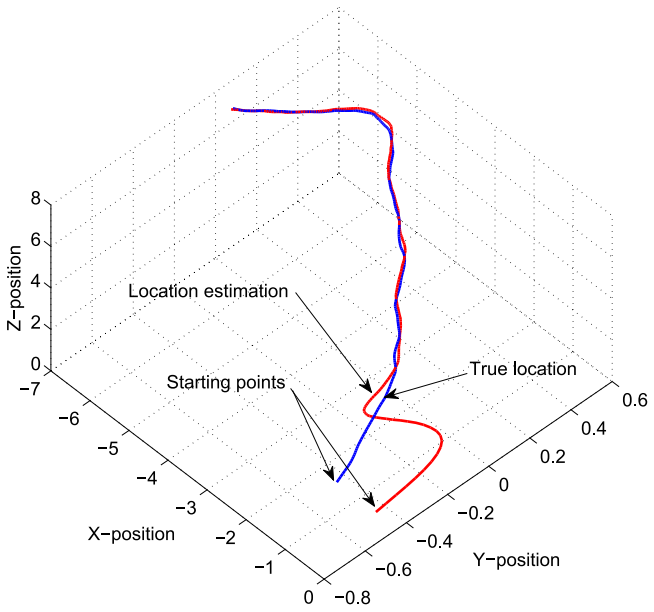

Fig. 7. Collaborative target tracking using the DQKF, showing the position of the target and its estimates along the X, Y, and Z axes. The initial values for the target position and its estimate are denoted as "starting points".

are used in their augmented formulation given by

$$\forall l \in \mathcal{N}: \begin{cases} \mathbf{x}^a_{n+1} = \mathbf{A}^a \mathbf{x}^a_n + \mathbf{B}^a_n \boldsymbol{\eta}^a_n \\ \mathbf{y}^a_{l,n} = \mathbf{H}^a \mathbf{x}^a_n + \boldsymbol{\omega}^a_{l,n} \end{cases} \tag{34}$$

with $\mathbf{A}^a = \text{block-diag}(\mathbf{A})$, $\mathbf{H}^a = \text{block-diag}(\mathbf{H})$, and $\mathbf{B}^a = \text{block-col}(\mathbf{B})$ representing block diagonal matrices that have $\mathbf{A}$, $\mathbf{H}$, and $\mathbf{B}$ as their diagonal elements respectively.

Remark 8: Note that since state evolution and observation noise have vanishing real components, i.e. $\Re(\eta_n) = \Re(\omega_n) = 0$, they will have non-vanishing pseudo-covariances and the augmented formulation becomes necessary in order fully incorporate the second-order information.

In the first simulation, the quaternion Kalman filter was implemented to track a maneuvering target, where $\Delta T = 0.04$ s, whereas η_n was a zero-mean unit variance quaternion Gaussian random variable with all its pseudo-covariances equal to -0.33, while for all nodes in the network the observation noise was selected as a zero-mean quaternion Gaussian variable with the variance of 0.009 and pseudo-covariances $c_{\omega\omega^i} = -0.007$, $c_{\omega\omega^j} = -0.001$, and $c_{\omega\omega^k} = -0.001$. The estimate of the location of the target, at the node denoted in Figure 1, obtained through the DQKF are shown in Figure 7; in addition, the steady-state MSE performance of each node is shown in Figure 8. Observe that the newly developed DQKF can accurately track the target and that the steady-state MSE of each node obtained through simulations closely follows those obtained through the analysis in Section IV-C.

We next considered the problem of tracking the position of a maneuvering target where the sensors can only measure the bearings of the target. Commonly referred to as bearings-only tracking, this problem is often encountered in passive radar or sonar tracking applications. Since none of the nodes have access

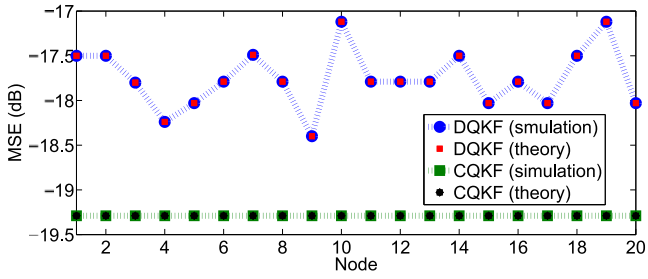

Fig. 8. Steady-state MSE performance of different nodes in collaborative target tracking implemented through the DQKF and CQKF.

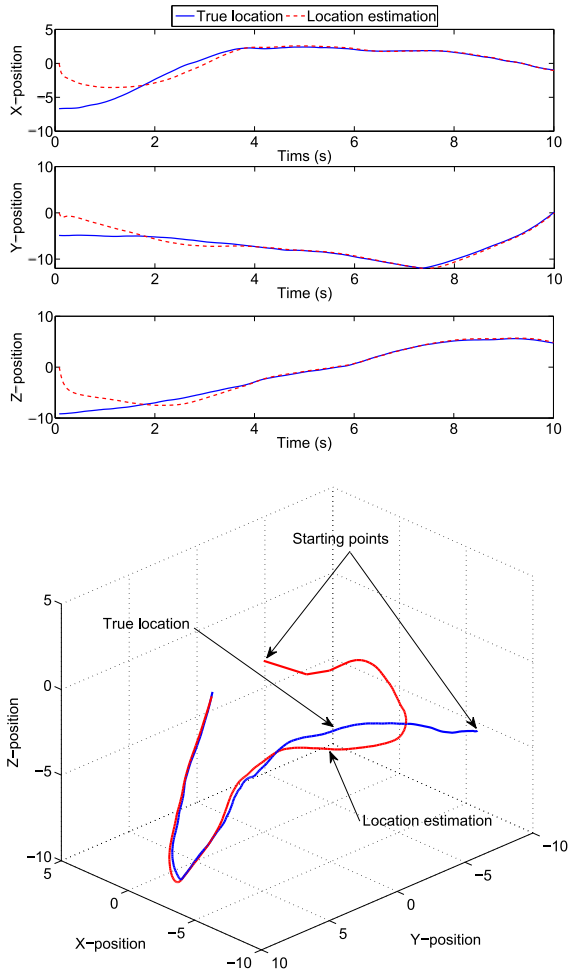

Fig. 9. Collaborative target tracking using bearings-only measurements. Position of the target and its estimates along the X, Y, and Z axes are shown respectively in the top three graphs, while the bottom graph shows the location of the target and its estimate in the three-dimensional space. The initial values for the target position and its estimate are denoted as "starting points".

to the range of the target, arriving at a unique solution using only the information available to one node is not possible. A solution to this problem is given in [26] using a quaternion Kalman filter that combines the observations of two sensors in order to locate the target through triangulation; however, the results are not generalizable for implementation over sensor networks. Taking into account that the developed DQKF operates akin to a CQKF that has access to observations from its neighboring nodes, in

the solution designed here, the proposed DQKF is implemented in the sensor network where the diffusion of local estimates is exploited to force the nodes to arrive at a unique solution, based on observations from all nodes in the network.

In the second simulation, we considered tracking the location of a target moving inside a $24 \times 24 \times 24$ cube using bearings-only measurements. The $24 \times 24 \times 24$ cube was sub-divided into 20 equal sized cubes each housing, at its center, a node of the network shown in Figure 1. The bearings-only measurements for every node in the network is given by

$$\forall l \in \mathcal{N} : y_{l,n} = \frac{L_n^{targ} - L_l^{sen}}{\left| L_n^{targ} - L_l^{sen} \right|} + \omega_{l,n}$$

where L_n^{targ} represents the location of the target at time instant n, L_l^{sen} denotes the location of node (sensor) l, and $\omega_{l,n}$ is the observation noise at node l at time instant n. Note that the state evolution equation remains the same as the one given in (32). The sampling interval was $\Delta T = 0.04$ s, with the state and observation noise statistics kept the same as in the previous simulation. The estimate of the location of the target at the node at $(4.4, 4.4, 4.4)$ is shown in Figure 9. Observe that the proposed algorithm was able to accurately track the location of the target.

VI. Conclusion

A distributed quaternion Kalman filter has been developed for distributed sequential state estimation in sensor networks. This has been achieved through decomposing the operations of the centralized quaternion Kalman filter in such a fashion that they can be performed by individual nodes in the network so that the final state vector estimate can be obtained by averaging local estimates obtained at each node. The proposed algorithm differs from existing distributed Kalman filtering techniques in that it does not require mixing coefficients for averaging local estimates. The proposed algorithm also takes into account the effect of averaging local estimates on the *a posteriori* estimate of the augmented covariance matrix of the augmented state vector estimation error. In addition, the concept has been expanded for application in multi-task networks. The performance of the developed algorithm has been analyzed and quantified through establishing a recursive expression for the estimation error. Finally, the developed algorithm has been used for estimating the fundamental frequency of three-phase power distribution networks and for collaborative target tracking.

References

[1] R. Olfati-Saber, "Distributed tracking for mobile sensor networks with information-driven mobility," in *Proc. Am. Control Conf.*, Jul. 2007, pp. 4604–4612.

[2] R. Olfati-Saber, "Distributed Kalman filtering for sensor networks," in *Proc. IEEE Int. Conf. Decision. Control*, 2007, pp. 5492–5498.

[3] D. E. Chang, S. C. Shawn, J. E. Marsden, and R. Olfati-Saber, "Collision avoidance for multiple agent systems," in *Proc. IEEE Int. Conf. Decision. Control*, Dec. 2003, pp. 539–543.

[4] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Automat. Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.

[5] E. Franco, R. Olfati-Saber, T. Parisini, and M. M. Polycarpou, "Distributed fault diagnosis using sensor networks and consensus-based filters," in *Proc. IEEE Int. Conf. Decision. Control*, Dec. 2006, pp. 386–391.

[6] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: Algorithms and theory," *IEEE Trans. Automat. Control*, vol. 51, no. 3, pp. 401–420, Mar. 2006.

[7] R. Olfati-Saber and P. Jalalkamali, "Coupled distributed estimation and control for mobile sensor networks," *IEEE Trans. Automat. Control*, vol. 57, no. 10, pp. 2609–2614, Oct. 2012.

[8] F. Cattivelli and A. H. Sayed, "Distributed nonlinear Kalman filtering with applications to wireless localization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2010, pp. 3522–3525.

[9] F. Cattivelli and A. H. Sayed, "Diffusion strategies for distributed Kalman filtering and smoothing," *IEEE Trans. Automat. Control*, vol. 55, no. 9, pp. 2069–2084, Sep. 2010.

[10] A. H. Sayed, "Adaptation, learning, and optimization over networks," *Found. Trends Mach. Learn.*, vol. 7, nos. 4/5, pp. 311–801, 2014.

[11] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1035–1048, Mar. 2010.

[12] C. Jianshu and A. H. Sayed, "Diffusion adaptation strategies for distributed optimization and learning over networks," *IEEE Trans. Signal Process.*, vol. 60, no. 8, pp. 4289–4305, Aug. 2012.

[13] P. Braca, P. Willett, K. LePage, S. Marano, and V. Matta, "Bayesian tracking in underwater wireless sensor networks with port-starboard ambiguity," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1864–1878, Apr. 2014.

[14] J. Chen, C. Richard, and A. H. Sayed, "Diffusion LMS over multitask networks," *IEEE Trans. Signal Process.*, vol. 63, no. 13, pp. 2733–3300, Jul. 2015.

[15] P. Ogren, E. Fiorelli, and N. E. Leonard, "Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment," *IEEE Trans. Automat. Control*, vol. 49, no. 8, pp. 1292–1302, Aug. 2004.

[16] S. Aeron, V. Saligrama, and D. V. Castanon, "Efficient sensor management policies for distributed target tracking in multihop sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 6, pp. 2562–2574, Jun. 2008.

[17] I. D. Schizas, G. Mateos, and G. B. Giannakis, "Distributed LMS for consensus-based in-network adaptive processing," *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2365–2382, Jun. 2009.

[18] S. Kanna, D. H. Dini, Y. Xia, S. Y. Hui, and D. P. Mandic, "Distributed widely linear Kalman filtering for frequency estimation in power networks," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 1, no. 1, pp. 45–57, Mar. 2015.

[19] S. Y. Tu and A. H. Sayed, "Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6217–6234, Dec. 2012.

[20] J. B. Kuipers, *Quaternions and Rotation Sequences: A Primer With Applications to Orbits, Aerospace and Virtual Reality*. Princeton, NJ, USA: Princeton Univ. Press, Aug. 2002.

[21] 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.

[22] 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.*, Jul. 2010, pp. 1–5.

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

[24] J. Via, 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.

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

[26] C. Jahanchahi and D. P. Mandic, "A class of quaternion Kalman filters," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 3, pp. 533–544, Mar. 2014.

[27] S. P. Talebi and D. P. Mandic, "A quaternion frequency estimator for three-phase power systems," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 2015, pp. 3956–3960.

[28] S. C. Pei and C. M. Cheng, "Color image processing by using binary quaternion-moment-preserving thresholding technique," *IEEE Trans. Image Process.*, vol. 8, no. 5, pp. 614–628, May 1999.

[29] C. E. Moxey, S. J. Sangwine, and T. A. Ell, "Hypercomplex correlation techniques for vector images," *IEEE Trans. Signal Process.*, vol. 51, no. 7, pp. 1941–1953, Jul. 2003.

[30] J. L. Crassidis, F. L. Markley, and Y. Cheng, "Survey of nonlinear attitude estimation methods," *J. Guid., Control, Dyn.*, vol. 30, no. 1, pp. 12–28, Jan. 2007.
[31] F. A. Tobar and D. P. Mandic, "Quaternion reproducing kernel Hilbert spaces: Existence and uniqueness conditions," *IEEE Trans. Inf. Theory*, vol. 60, no. 9, pp. 5736–5749, Sep. 2014.
[32] T. K. Paul and T. Ogunfunmi, "A kernel adaptive algorithm for quaternion-valued inputs," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 10, pp. 2422–2439, Oct. 2015.
[33] C. C. Took, G. Strbac, K. Aihara, and D. P. Mandic, "Quaternion-valued short-term joint forecasting of three-dimensional wind and atmospheric parameters," *Renewable Energy*, vol. 36, no. 6, pp. 1754–1760, Jun. 2011.
[34] C. Jahanchahi and D. P. Mandic, "An adaptive diffusion quaternion LMS algorithm for distributed networks of 3D and 4D vector sensors," in *Proc. 21st Eur. Signal Process. Conf.*, Sep. 2013, pp. 1–5.
[35] T. A. Ell and S. J. Sangwine, "Quaternion involutions and anti-involutions," *Comput. Math. Appl.*, vol. 53, no. 1, pp. 137–143, Jan. 2007.
[36] S. Said, N. Le Bihan, and S. J. Sangwine, "Fast complexified quaternion Fourier transform," *IEEE Trans. Signal Process.*, vol. 56, no. 4, pp. 1522–1531, Apr. 2008.
[37] K. Abdel-Khalek, "Quaternion analysis," Dipartimento di Fisica, Universita de Lecce, Lecce, Italy, Tech. Rep., 1996, [Online]. Available at: http://arxiv.org/abs/hep-th/9607152v2.
[38] P. J. Schreier and L. L. Scharf, *Statistical Signal Processing of Complex-Valued Data: The Theory of Improper and Noncircular Signals*, Cambridge, U.K.: Cambridge Univ. Press, 2010.
[39] B. Picinbono and P. Chevalier, "Widely linear estimation with complex data," *IEEE Trans. Signal Process.*, vol. 43, no. 8, pp. 2030–2033, Aug. 1995.
[40] T. Kailath, A. H. Sayed, and B. Hassibi, *Linear Estimation*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2000.
[41] J. Stange, N. Loginova, and T. Dickhaus, "Computing and approximating multivariate chi-square probabilities," *J. Statist. Comput. Simul.*, vol. 86, no. 6, pp. 1233–1247, 2016.
[42] T. Dickhaus and T. Royen, "A survey on multivariate chi-square distributions and their applications in testing multiple hypotheses," *Statistics*, vol. 49, no. 2, pp. 427–454, 2015.
[43] E. Clarke, *Circuit Analysis of A.C. Power Systems*. New York, NY, USA: Wiley, 1943.
[44] R. K. Varma, R. M. Mathur, G. J. Rogers, and P. Kundur, "Modeling effects of system frequency variation in long-term stability studies," *IEEE Trans. Power Syst.*, vol. 11, no. 2, pp. 827–832, May 1996.
[45] A. Von Jouanne and B. Banerjee, "Assessment of voltage unbalance," *IEEE Trans. Power Del.*, vol. 16, no. 4, pp 782–790, Oct. 2001.
[46] D. H. Dini and D. P. Mandic, "Widely linear modeling for frequency estimation in unbalanced three-phase power systems," *IEEE Trans. Instrum. Meas.*, vol. 62, no. 2, pp. 353–363, Feb. 2013.
[47] P. K. Dash, A. K. Pradhan, and G. Panda, "Frequency estimation of distorted power system signals using extended complex Kalman filter," *IEEE Trans. Power Del.*, vol. 14, no. 3, pp. 761–766, Jul. 1999.

Sayed Pouria Talebi received the B.Sc. degree in electrical engineering from Isfahan University, Isfahan, Iran and the M.Sc. degree from the University of Leeds, Leeds, U.K. He is currently working toward the Ph.D. degree at Imperial College London, London, U.K. His research interests include quaternion-valued and distributed signal processing.

Sithan Kanna (S'13) received the M.Eng. degree in electrical and electronic engineering with management from Imperial College London, London, U.K., in 2012. He is currently working the Ph.D. degree in adaptive signal processing through a full scholarship by The Rector's Fund at Imperial College London. His current research interests also include complex-valued statistical signal processing and frequency estimation in the smart grid.

Danilo P. Mandic (M'99–SM'03–F'12) is a Professor of signal processing with Imperial College London, London, U.K., where he has been involved in nonlinear adaptive signal processing and nonlinear dynamics. He has been a Guest Professor with Katholieke Universiteit Leuven, Leuven, Belgium, the Tokyo University of Agriculture and Technology, Tokyo, Japan, and Westminster University, London, U.K., and a Frontier Researcher with RIKEN, Wako, Japan. He has two research monographs titled *Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability* (West Sussex, U.K.: Wiley, 2001) and *Complex Valued Nonlinear Adaptive Filters: Noncircularity, Widely Linear and Neural Models* (West Sussex, U.K.: Wiley, 2009), an edited book titled *Signal Processing Techniques for Knowledge Extraction and Information Fusion* (New York, NY, USA: Springer, 2008), and more than 200 publications on signal and image processing. He has been a member of the IEEE Technical Committee on Signal Processing Theory and Methods and an Associate Editor of the IEEE SIGNAL PROCESSING MAGAZINE, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and the IEEE TRANSACTIONS ON NEURAL NETWORKS. He has produced award winning papers and products resulting from his collaboration with the industry.