LEARNING ALGORITHMS IN QUATERNION NEURAL NETWORKS USING GHR CALCULUS

D. Xu[*], *L. Zhang*[†], *H. Zhang*[‡]

Abstract: One difficulty for quaternion neural networks (QNNs) is that quaternion nonlinear activation functions are usually non-analytic and thus quaternion derivatives cannot be used. In this paper, we derive the quaternion gradient descent, approximated quaternion Gauss-Newton and quaternion Levenberg-Marquardt algorithms for feedforward QNNs based on the GHR calculus, which is suitable for analytic and non-analytic quaternion functions. Meanwhile, we solve a widely linear quaternion least squares problem in the derivation of quaternion Gauss-Newton algorithm, which is more general than the usual least squares problem. A rigorous analysis of the convergence of the proposed algorithms is provided. Simulations on the prediction of benchmark signals support the approach.

Key words: *quaternion neural networks, non-analytic quaternion activation functions, GHR calculus, learning algorithms, convergence*

Received: November 18, 2014 **DOI:** 10.14311/NNW.2017.27.014
Revised and accepted: May 4, 2017

1. Introduction

In recent years, quaternion neural networks have been applied to some engineering problems, such as control problems [1], color image compression [9], inertial body sensors [19], and wind profile modeling [11, 14, 16]. In these applications, quaternions have been allowed for a reduction in the number of parameters and operations involved. The characteristic of QNNs is that their inputs/outputs, parameters and activation functions are quaternion-valued and they can directly process quaternion-valued data. One of the difficulties in constructing QNNs is about the choice of the quaternion nonlinear activation functions. A typical approach, 'splitting', uses a real-valued activation function to process each component of quaternion value. However, such split-type activation functions are often non-analytic according to quaternion analysis [4] and therefore quaternion derivatives cannot be used. To relax this constraint, the so-called Cauchy-Riemann-Fueter

[*]Dongpo Xu; School of Mathematics and Statistics, Northeast Normal University, Changchun 130024, P.R. China, E-mail: dongpoxu@gmail.com
[†]Lina Zhang; College of Science, Harbin Engineering University, Harbin 150001, P.R. China
[‡]Huisheng Zhang; Department of Mathematics, Dalian Maritime University, Dalian 116026, P.R. China

(CRF) condition [13] is often adopted for the quaternion functions. However, the even polynomial functions do not satisfy the CRF condition. Another class of analyticity for the quaternion functions has been developed in [8] and is called 'local analyticity', but the product and composition of two local analytic functions are generally not local analytic. Therefore, it is necessary to define a general quaternion derivatives for quaternion analytic and non-analytic functions. The recently proposed GHR calculus [21, 22] precisely satisfies this requirement and comprises a novel product rule and chain rule, which can be considered as a generalization of the Wirtinger calculus [2, 7, 18] to the quaternion field.

In this paper, we focus on the derivation of quaternion learning algorithms and the convergence analysis of the algorithms. Quaternion gradient and direction of steepest descent based on GHR calculus [23,24] are used to construct the quaternion gradient algorithms for feedward QNNs. A widely linear least squares problem $\min_{\mathbf{q}} ||\mathbf{b} - (\mathbf{Aq} + \mathbf{Bq}^i + \mathbf{Cq}^j + \mathbf{Dq}^k)||$ is found in the quaternion field, which is more general than the $\min_{\mathbf{q}} ||\mathbf{b} - \mathbf{Aq}||$. Solving the widely linear least squares problem plays an important role in the derivation of quaternion Gauss-Newton and Levenberg-Marquardt algorithms. An exact form of quaternion Gauss-Newton update is too complicated, so we propose an approximated Gauss-Newton update to reduce the computational cost. An approximated quaternion Levenberg-Marquardt algorithm is also derived in a similar manner. All computations regarding to the derivation of the algorithms are carried out directly in the quaternion field without transforming the problem to the real domain. Finally, we prove the convergence of the proposed quaternion learning algorithms under suitable conditions.

2. Preliminaries

2.1 The GHR calculus

The GHR calculus is derived using a generalized basis $\{1, i^\mu, j^\mu, k^\mu\}$, where $q^\mu = \mu q \mu^{-1}$ $(\mu \neq 0, \mu \in \mathbb{H})$ is the quaternion rotation. Observe that $i^\mu i^\mu = j^\mu j^\mu = k^\mu k^\mu = i^\mu j^\mu k^\mu = -1$, and hence this new basis also consists of imaginary unit vectors. Using this new basis, we obtain the expressions for the GHR calculus

Definition 1 (The GHR derivatives [21, 22])**.** Let $f : \mathbb{H} \to \mathbb{H}$. Then the GHR derivatives of $f(q)$ with respect to q^μ and $q^{\mu *}$ $(\mu \neq 0, \mu \in \mathbb{H})$ are defined as $\frac{\partial f}{\partial q^\mu} = \frac{1}{4}\left(\frac{\partial f}{\partial q_a} - \frac{\partial f}{\partial q_b} i^\mu - \frac{\partial f}{\partial q_c} j^\mu - \frac{\partial f}{\partial q_d} k^\mu\right)$, $\frac{\partial f}{\partial q^{\mu *}} = \frac{1}{4}\left(\frac{\partial f}{\partial q_a} + \frac{\partial f}{\partial q_b} i^\mu + \frac{\partial f}{\partial q_c} j^\mu + \frac{\partial f}{\partial q_d} k^\mu\right)$, where $q = q_a + iq_b + jq_c + kq_d$, $q_a, q_b, q_c, q_d \in \mathbb{R}$, and $\frac{\partial f}{\partial q_a}, \frac{\partial f}{\partial q_b}, \frac{\partial f}{\partial q_c}, \frac{\partial f}{\partial q_d} \in \mathbb{H}$ are the partial derivatives of f with respect to q_a, q_b, q_c and q_d, respectively.

Some useful rules of the GHR derivatives [21] are summarized as follows:

$$\begin{aligned} &\text{Product rule:} \quad \frac{\partial(fg)}{\partial q} = f\frac{\partial g}{\partial q} + \frac{\partial(fg)}{\partial q^g} g, \ \frac{\partial(fg)}{\partial q^*} = f\frac{\partial g}{\partial q^*} + \frac{\partial(fg)}{\partial q^{g*}} g, \\ &\text{Chain rule:} \quad \frac{\partial f(g(q))}{\partial q} = \sum_{\nu \in \{1,i,j,k\}} \frac{\partial f}{\partial g^\nu}\frac{\partial g^\nu}{\partial q}, \ \frac{\partial f(g(q))}{\partial q^*} = \sum_{\nu \in \{1,i,j,k\}} \frac{\partial f}{\partial g^\nu}\frac{\partial g^\nu}{\partial q^*}, \end{aligned} \tag{1}$$

$$\text{Rotation rule:} \quad \left(\frac{\partial f}{\partial q}\right)^{\nu} = \frac{\partial f^{\nu}}{\partial q^{\nu}}, \quad \left(\frac{\partial f}{\partial q^{*}}\right)^{\nu} = \frac{\partial f^{\nu}}{\partial q^{\nu *}}. \tag{2}$$

2.2 Widely Linear Quaternion Model

The existing (strictly linear) estimation in the quaternion domain is given by

$$\hat{y} = \mathbf{w}^{\mathrm{T}}\mathbf{x}, \tag{3}$$

where $\mathbf{x} = \mathbf{x}_a + i\mathbf{x}_b + j\mathbf{x}_c + k\mathbf{x}_d$. Observe that for all the quaternion components

$$\hat{y}_\eta = \mathrm{E}[y_\eta|\mathbf{x}_a, \mathbf{x}_b, \mathbf{x}_c, \mathbf{x}_d], \quad \eta \in \{a, b, c, d\}, \tag{4}$$

and using the involutions in [5], we can express the components of a quaternion via its involutions e.g. $\mathbf{x}_a = \frac{1}{4}(\mathbf{x} + \mathbf{x}^i + \mathbf{x}^j + \mathbf{x}^k)$, leading to

$$\hat{y}_\eta = \mathrm{E}[y_\eta|\mathbf{x}_a, \mathbf{x}^i, \mathbf{x}^j, \mathbf{x}^k] \text{ and } \hat{y} = \mathrm{E}[y|\mathbf{x}_a, \mathbf{x}^i, \mathbf{x}^j, \mathbf{x}^k]. \tag{5}$$

In other words, to capture the full second-order information available, we should use the quaternion *widely linear* model

$$\hat{y} = \mathbf{u}^{\mathrm{T}}\mathbf{x} + \mathbf{v}^{\mathrm{T}}\mathbf{x}^i + \mathbf{g}^{\mathrm{T}}\mathbf{x}^j + \mathbf{h}^{\mathrm{T}}\mathbf{x}^k. \tag{6}$$

Current statistical signal processing in $\mathbb{H}$ is largely based on strictly linear models, drawing upon the covariance matrix $\mathbf{R} = E[\mathbf{x}\mathbf{x}^{\mathrm{H}}]$. However, to model both the second-order circular (proper) and second-order noncircular (improper) signals, based on (6), we need to employ the augmented covariance matrix, given by [6, 15, 17, 20]

$$\mathbf{R}^a = \begin{pmatrix} \mathbf{R} & \mathbf{P} & \mathbf{S} & \mathbf{T} \\ \mathbf{P}^i & \mathbf{R}^i & \mathbf{T}^i & \mathbf{S}^i \\ \mathbf{S}^j & \mathbf{T}^j & \mathbf{R}^j & \mathbf{P}^j \\ \mathbf{T}^k & \mathbf{S}^k & \mathbf{P}^k & \mathbf{R}^k \end{pmatrix}. \tag{7}$$

3. Learning algorithms for QVNNs

3.1 Quaternion Gradient Descent algorithm

We consider a single hidden layer QVNN for convenience. The forward equations for signal passing through the network are as follows

$$\mathbf{y} = \mathbf{V}\mathbf{x} + \mathbf{a}, \ \mathbf{h} = \phi(\mathbf{y}), \ \mathbf{v} = \mathbf{W}\mathbf{h} + \mathbf{b}, \ \mathbf{g} = \phi(\mathbf{v}), \tag{8}$$

where $\mathbf{x}$ is the input signal and $\mathbf{h}, \mathbf{g}$ are the output at hidden and output layer, respectively. $\mathbf{V}, \mathbf{W}$ are the weight matrices associated with hidden and output layer neurons, $\mathbf{a}, \mathbf{b}$ are the biases to the hidden and output layer neurons, and ϕ is the activation function having real partial derivatives. The network error produced at the output layer is defined by $\mathbf{e} = \mathbf{d} - \mathbf{g}$, where $\mathbf{d}$ denotes the desired output. Then the gradient descent algorithm minimizes a real-valued loss function

$$\ell = \|\mathbf{e}\|^2 = \mathbf{e}^{\mathrm{H}}\mathbf{e}. \tag{9}$$

From [23, 24], the quaternion gradient of error function is given by

$$\nabla_{\mathbf{q}^*}\ell = \left(\frac{\partial \ell}{\partial \mathbf{q}^*}\right)^{\mathrm{T}} = \left(\frac{\partial \ell}{\partial \mathbf{q}}\right)^{\mathrm{H}}. \tag{10}$$

Using the chain rule (1) and rotation rule (2), we have

$$\frac{\partial \ell}{\partial \mathbf{q}} = \sum_{\mu\in\{1,i,j,k\}} \frac{\partial \|\mathbf{e}\|^2}{\partial \mathbf{e}^\mu}\frac{\partial\ \mathbf{e}^\mu}{\partial \mathbf{q}} = -\sum_{\mu\in\{1,i,j,k\}} \frac{\partial \|\mathbf{e}\|^2}{\partial \mathbf{e}^\mu}(\mathbf{J}_{\mathbf{q}^\mu})^\mu, \tag{11}$$

where $\mathbf{J}_{\mathbf{q}^\mu} \triangleq \frac{\partial \mathbf{g}}{\partial \mathbf{q}^\mu}$ is the Jacobian matrix of $\mathbf{g}$, and the derivative of $\|\mathbf{g}\|^2$ is a vector version of the term $\frac{\partial |q|^2}{\partial q^\mu}\mu$ in [23], given by

$$\frac{\partial \|\mathbf{e}\|^2}{\partial \mathbf{e}^\mu} = \frac{1}{2}\left(\mathbf{e}^\mu\right)^{\mathrm{H}}. \tag{12}$$

Substituting (12) and (11) into (10), we arrive at

$$\nabla_{\mathbf{q}^*}\ell = -\frac{1}{2}\sum_{\mu\in\{1,i,j,k\}} \left(\mathbf{J}^{\mathrm{H}}_{\mathbf{q}^\mu}\mathbf{e}\right)^\mu. \tag{13}$$

Now, we derive the gradient descent algorithm in explicit form for the QVNN

$$\mathbf{J}_{\mathbf{y}} = \frac{\partial \mathbf{g}}{\partial \mathbf{y}} = \sum_{\mu\in\{1,i,j,k\}} \frac{\partial \mathbf{g}}{\partial \mathbf{v}^\mu}\frac{\partial \mathbf{v}^\mu}{\partial \mathbf{y}} = \sum_{\mu\in\{1,i,j,k\}} \Lambda_{\mathbf{v}^\mu}\mathbf{W}^\mu(\Lambda_{\mathbf{y}^\mu})^\mu, \tag{14}$$

where $\Lambda_{\mathbf{v}^\mu}$ denotes the Jacobian of $\mathbf{g}$ with respect to $\mathbf{v}^\mu$. From $\mathbf{J}_{\mathbf{h}^\mu} = \Lambda_{\mathbf{v}^\mu}\mathbf{W}^\mu$, (14) can be rewritten as

$$\mathbf{J}_{\mathbf{y}} = \sum_{\mu\in\{1,i,j,k\}} \mathbf{J}_{\mathbf{h}^\mu}(\Lambda_{\mathbf{y}^\mu})^\mu. \tag{15}$$

We note from (8) that $\mathbf{J}_{\mathbf{y}} = \mathbf{J}_{\mathbf{a}}$ and $\mathbf{J}_{\mathbf{v}} = \mathbf{J}_{\mathbf{b}}$. Thus update rules for the biases at hidden and output layer are

$$\begin{aligned}
\Delta\mathbf{a} &= \alpha\left(\mathbf{J}^{\mathrm{H}}_{\mathbf{a}}\mathbf{e} + (\mathbf{J}^{\mathrm{H}}_{\mathbf{a}^i}\mathbf{e})^i + (\mathbf{J}^{\mathrm{H}}_{\mathbf{a}^j}\mathbf{e})^j + (\mathbf{J}^{\mathrm{H}}_{\mathbf{a}^k}\mathbf{e})^k\right),\\
\Delta\mathbf{b} &= \alpha\left(\mathbf{J}^{\mathrm{H}}_{\mathbf{b}}\mathbf{e} + (\mathbf{J}^{\mathrm{H}}_{\mathbf{b}^i}\mathbf{e})^i + (\mathbf{J}^{\mathrm{H}}_{\mathbf{b}^j}\mathbf{e})^j + (\mathbf{J}^{\mathrm{H}}_{\mathbf{b}^k}\mathbf{e})^k\right),
\end{aligned} \tag{16}$$

where $\alpha > 0$ is the learning rate. Using the chain rule (1), the update rules for hidden and output layer weight matrices are given by

$$\Delta\mathbf{V} = \Delta\mathbf{a}\mathbf{x}^{\mathrm{H}}, \Delta\mathbf{W} = \Delta\mathbf{b}\mathbf{x}^{\mathrm{H}}. \tag{17}$$

3.2 Quaternion Gauss-Newton algorithm

The Gauss-Newton algorithm can be seen as a modification of Newton's method for finding a minimum of a function. It has the advantage of not requiring the computation of second order derivatives of the function. In the QVNN, the linearized model of network output $\mathbf{g}(\mathbf{q})$ around $\mathbf{q}$ is given by [24]

$$\mathbf{g}(\mathbf{q}+\Delta\mathbf{q}) \approx \mathbf{g}(\mathbf{q}) + \mathbf{J}_{\mathbf{q}}\Delta\mathbf{q} + \mathbf{J}_{\mathbf{q}^i}\Delta\mathbf{q}^i + \mathbf{J}_{\mathbf{q}^j}\Delta\mathbf{q}^j + \mathbf{J}_{\mathbf{q}^k}\Delta\mathbf{q}^k. \tag{18}$$

The error associated with the linearized model is

$$\hat{\mathbf{e}} = \mathbf{e} - (\mathbf{J}_{\mathbf{q}}\Delta\mathbf{q} \ + \mathbf{J}_{\mathbf{q}^i}\Delta\mathbf{q}^i + \mathbf{J}_{\mathbf{q}^j}\Delta\mathbf{q}^j + \mathbf{J}_{\mathbf{q}^k}\Delta\mathbf{q}^k), \tag{19}$$

where $\hat{\mathbf{e}} = \mathbf{d} - \mathbf{g}(\mathbf{q} + \Delta\mathbf{q})$. The task is to find $\Delta\mathbf{q}$ to minimize the sum of squares of the right-hand side of (19), i.e.,

$$\min_{\mathbf{q}} ||\mathbf{e} - (\mathbf{J}_{\mathbf{q}}\Delta\mathbf{q} \ + \mathbf{J}_{\mathbf{q}^i}\Delta\mathbf{q}^i + \mathbf{J}_{\mathbf{q}^j}\Delta\mathbf{q}^j + \mathbf{J}_{\mathbf{q}^k}\Delta\mathbf{q}^k)||, \tag{20}$$

which is the widely linear least squares problem $\min_{\mathbf{q}} ||\mathbf{b}-(\mathbf{A}\mathbf{q}+\mathbf{B}\mathbf{q}^i+\mathbf{C}\mathbf{q}^j+\mathbf{D}\mathbf{q}^k)||$ instead of the well known problem $\min_{\mathbf{q}} ||\mathbf{b} - \mathbf{A}\mathbf{q}||$. The problem $\min_{\mathbf{q}} ||\mathbf{b} - \mathbf{A}\mathbf{q}||$ can be solved from the normal equation $\mathbf{A}^{\mathrm{H}}\mathbf{A}\mathbf{q} = \mathbf{A}^{\mathrm{H}}\mathbf{b}$, however, the normal equation of the widely linear least squares problem remains unknown. In the next proposition, we solve this problem.

Lemma 1. *Let* $\mathbf{A}, \mathbf{B}, \mathbf{C}$ *and* $\mathbf{D}$ *be arbitrary quaternion matrices of same dimension. Then a solution to the widely linear least squares problem,* $\min_{\mathbf{q}} ||\mathbf{b} - (\mathbf{A}\mathbf{q} + \mathbf{B}\mathbf{q}^i + \mathbf{C}\mathbf{q}^j + \mathbf{D}\mathbf{q}^k)||$*, is given by the following normal equation*

$$\mathbf{Q}^{\mathrm{H}} \begin{pmatrix} \mathbf{b} \\ \mathbf{b}^i \\ \mathbf{b}^j \\ \mathbf{b}^k \end{pmatrix} = \mathbf{Q}^{\mathrm{H}}\mathbf{Q} \begin{pmatrix} \mathbf{q} \\ \mathbf{q}^i \\ \mathbf{q}^j \\ \mathbf{q}^k \end{pmatrix}, \text{ where } \mathbf{Q} = \begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{C} & \mathbf{D} \\ \mathbf{B}^i & \mathbf{A}^i & \mathbf{D}^i & \mathbf{C}^i \\ \mathbf{C}^j & \mathbf{D}^j & \mathbf{A}^j & \mathbf{B}^j \\ \mathbf{D}^k & \mathbf{C}^k & \mathbf{B}^k & \mathbf{A}^k \end{pmatrix}. \tag{21}$$

Proof. From [24], we know that the augmented quaternion vector is related with the dual quadrivariate real vector by the transformation matrix $\mathbf{J}$, while satisfying

$$\frac{1}{4}\mathbf{J}\mathbf{J}^{\mathrm{H}} = \frac{1}{4}\begin{bmatrix} \mathbf{I} & i\mathbf{I} & j\mathbf{I} & k\mathbf{I} \\ \mathbf{I} & i\mathbf{I} & -j\mathbf{I} & -k\mathbf{I} \\ \mathbf{I} & -i\mathbf{I} & j\mathbf{I} & -k\mathbf{I} \\ \mathbf{I} & -i\mathbf{I} & -j\mathbf{I} & k\mathbf{I} \end{bmatrix}\begin{bmatrix} \mathbf{I} & \mathbf{I} & \mathbf{I} & \mathbf{I} \\ -i\mathbf{I} & -i\mathbf{I} & i\mathbf{I} & i\mathbf{I} \\ -j\mathbf{I} & j\mathbf{I} & -j\mathbf{I} & j\mathbf{I} \\ -k\mathbf{I} & k\mathbf{I} & k\mathbf{I} & -k\mathbf{I} \end{bmatrix} = \mathbf{I}. \tag{22}$$

The residual and its involutions associated to the least squares problem are

$$\begin{aligned} \mathbf{r} &= \mathbf{b} - (\mathbf{A}\mathbf{q} + \mathbf{B}\mathbf{q}^i + \mathbf{C}\mathbf{q}^j + \mathbf{D}\mathbf{q}^k), \\ \mathbf{r}^i &= \mathbf{b}^i - (\mathbf{A}^i\mathbf{q}^i + \mathbf{B}^i\mathbf{q} + \mathbf{C}^i\mathbf{q}^k + \mathbf{D}^i\mathbf{q}^j), \\ \mathbf{r}^j &= \mathbf{b}^j - (\mathbf{A}^j\mathbf{q}^j + \mathbf{B}^j\mathbf{q}^k + \mathbf{C}^j\mathbf{q} + \mathbf{D}^j\mathbf{q}^i), \\ \mathbf{r}^k &= \mathbf{b}^k - (\mathbf{A}^k\mathbf{q}^k + \mathbf{B}^k\mathbf{q}^j + \mathbf{C}^k\mathbf{q}^i + \mathbf{D}^k\mathbf{q}). \end{aligned} \tag{23}$$

Combining the above equations to form a matrix equation and applying the linear transformation $\mathbf{J}$, the problem can be transformed to real coordinate system, yields

$$\mathbf{J}^{\mathrm{H}} \begin{pmatrix} \mathbf{r} \\ \mathbf{r}^i \\ \mathbf{r}^j \\ \mathbf{r}^k \end{pmatrix} = \mathbf{J}^{\mathrm{H}} \begin{pmatrix} \mathbf{b} \\ \mathbf{b}^i \\ \mathbf{b}^j \\ \mathbf{b}^k \end{pmatrix} - \mathbf{J}^{\mathrm{H}}\mathbf{Q}(\frac{1}{4}\mathbf{J}\mathbf{J}^{\mathrm{H}}) \begin{pmatrix} \mathbf{q} \\ \mathbf{q}^i \\ \mathbf{q}^j \\ \mathbf{q}^k \end{pmatrix}, \quad \left[\frac{1}{4}\mathbf{J}\mathbf{J}^{\mathrm{H}} = \mathbf{I}\right]. \tag{24}$$

It can be shown that $\frac{1}{4}\mathbf{J}^{\mathrm{H}}\mathbf{Q}\mathbf{J}$ is a real-valued matrix and (24) operates in the real domain. Thus, the normal equation for the least squares problem (24) is

$$\frac{1}{4}\mathbf{J}^{\mathrm{H}}\mathbf{Q}^{\mathrm{H}}\mathbf{J}\mathbf{J}^{\mathrm{H}}\begin{pmatrix}\mathbf{b}\\ \mathbf{b}^{i}\\ \mathbf{b}^{j}\\ \mathbf{b}^{k}\end{pmatrix} = \frac{1}{4}\mathbf{J}^{\mathrm{H}}\mathbf{Q}^{\mathrm{H}}\mathbf{J}\mathbf{J}^{\mathrm{H}}\mathbf{Q}(\frac{1}{4}\mathbf{J}\mathbf{J}^{\mathrm{H}})\begin{pmatrix}\mathbf{q}\\ \mathbf{q}^{i}\\ \mathbf{q}^{j}\\ \mathbf{q}^{k}\end{pmatrix}. \tag{25}$$

Since $\frac{1}{4}\mathbf{J}\mathbf{J}^{\mathrm{H}} = \mathbf{I}$ and invertibility of $\mathbf{J}^{\mathrm{H}}$, equation (25) immediately yields the following quaternion normal equation

$$\mathbf{Q}^{\mathrm{H}}\begin{pmatrix}\mathbf{b}\\ \mathbf{b}^{i}\\ \mathbf{b}^{j}\\ \mathbf{b}^{k}\end{pmatrix} = \mathbf{Q}^{\mathrm{H}}\mathbf{Q}\begin{pmatrix}\mathbf{q}\\ \mathbf{q}^{i}\\ \mathbf{q}^{j}\\ \mathbf{q}^{k}\end{pmatrix}. \tag{26}$$

This completes the proof of Lemma 1. □

According to Lemma 1, the least squares solution to (20) gives the following Gauss-Newton update rule

$$\mathbf{Q}^{\mathrm{H}}\mathbf{Q}\begin{pmatrix}\Delta\mathbf{q}\\ \Delta\mathbf{q}^{i}\\ \Delta\mathbf{q}^{j}\\ \Delta\mathbf{q}^{k}\end{pmatrix} = \mathbf{Q}^{\mathrm{H}}\begin{pmatrix}\mathbf{e}\\ \mathbf{e}^{i}\\ \mathbf{e}^{j}\\ \mathbf{e}^{k}\end{pmatrix}, \tag{27}$$

where

$$\mathbf{Q} = \left(\begin{array}{c|ccc} \mathbf{J}_{\mathbf{q}} & \mathbf{J}_{\mathbf{q}^{i}} & \mathbf{J}_{\mathbf{q}^{j}} & \mathbf{J}_{\mathbf{q}^{k}} \\ \hline (\mathbf{J}_{\mathbf{q}^{i}})^{i} & (\mathbf{J}_{\mathbf{q}})^{i} & (\mathbf{J}_{\mathbf{q}^{k}})^{i} & (\mathbf{J}_{\mathbf{q}^{j}})^{i} \\ (\mathbf{J}_{\mathbf{q}^{j}})^{j} & (\mathbf{J}_{\mathbf{q}^{k}})^{j} & (\mathbf{J}_{\mathbf{q}})^{j} & (\mathbf{J}_{\mathbf{q}^{i}})^{j} \\ (\mathbf{J}_{\mathbf{q}^{k}})^{k} & (\mathbf{J}_{\mathbf{q}^{j}})^{k} & (\mathbf{J}_{\mathbf{q}^{i}})^{k} & (\mathbf{J}_{\mathbf{q}})^{k} \end{array}\right) = \begin{pmatrix}\mathbf{Q}_{11} & \mathbf{Q}_{12}\\ \mathbf{Q}_{21} & \mathbf{Q}_{22}\end{pmatrix}. \tag{28}$$

Let

$$\mathbf{H} = \mathbf{Q}^{\mathrm{H}}\mathbf{Q} = \begin{pmatrix}\mathbf{H}_{11} & \mathbf{H}_{12}\\ \mathbf{H}_{21} & \mathbf{H}_{22}\end{pmatrix}, \quad \text{where } \mathbf{H}_{11} = \sum_{\mu\in\{1,i,j,k\}} \left(\mathbf{J}_{\mathbf{q}^{\mu}}^{\mathrm{H}}\mathbf{J}_{\mathbf{q}^{\mu}}\right)^{\mu}. \tag{29}$$

If $\mathbf{H}_{11}$ is invertible, then use the Banach Schwartz inversion formula for the inverse of a non-singular partitioned matrix [25], yields

$$\begin{pmatrix}\Delta\mathbf{q}\\ \Delta\mathbf{q}^{i}\\ \Delta\mathbf{q}^{j}\\ \Delta\mathbf{q}^{k}\end{pmatrix} = \begin{pmatrix}\mathbf{H}_{11}^{-1}+\mathbf{L}\mathbf{T}^{-1}\mathbf{U} & -\mathbf{L}\mathbf{T}^{-1}\\ -\mathbf{T}^{-1}\mathbf{U} & \mathbf{T}^{-1}\end{pmatrix}\mathbf{Q}^{\mathrm{H}}\begin{pmatrix}\mathbf{e}\\ \mathbf{e}^{i}\\ \mathbf{e}^{j}\\ \mathbf{e}^{k}\end{pmatrix}, \tag{30}$$

where $\mathbf{L} = \mathbf{H}_{11}^{-1}\mathbf{H}_{12}$, $\mathbf{U} = \mathbf{H}_{21}\mathbf{H}_{11}^{-1}$ and $\mathbf{T} = (\mathbf{H}/\mathbf{H}_{11})$ is the Schur complement of $\mathbf{H}_{11}$ in $\mathbf{H}$. Thus, the quaternion Gauss-Newton update rule is given by

$$\begin{aligned}\Delta\mathbf{q} = &\left((\mathbf{H}_{11}^{-1}+\mathbf{L}\mathbf{T}^{-1}\mathbf{U})\mathbf{Q}_{11}^{\mathrm{H}}+\mathbf{L}\mathbf{T}^{-1}\mathbf{Q}_{12}^{\mathrm{H}}\right)\mathbf{e}\\ &+\left((\mathbf{H}_{11}^{-1}+\mathbf{L}\mathbf{T}^{-1}\mathbf{U})\mathbf{Q}_{21}^{\mathrm{H}}+\mathbf{L}\mathbf{T}^{-1}\mathbf{Q}_{22}^{\mathrm{H}}\right)\begin{pmatrix}\mathbf{e}^{i}\\ \mathbf{e}^{j}\\ \mathbf{e}^{k}\end{pmatrix}.\end{aligned} \tag{31}$$

To avoid computing the inverse of the Schur complement $\mathbf{T}$ and to provide a simplification, the Guass-Newton method in (31) can be approximated as

$$\Delta\mathbf{q} \approx \mathbf{H}_{11}^{-1}\left(\mathbf{J}_{\mathbf{q}}^{\mathrm{H}}\mathbf{e} + (\mathbf{J}_{\mathbf{q}^i}^{\mathrm{H}}\mathbf{e})^i + (\mathbf{J}_{\mathbf{q}^j}^{\mathrm{H}}\mathbf{e})^j + (\mathbf{J}_{\mathbf{q}^k}^{\mathrm{H}}\mathbf{e})^k\right) = (\mathbf{G}_{\mathbf{q}}^{\mathrm{H}}\mathbf{G}_{\mathbf{q}})^{-1}\mathbf{G}_{\mathbf{q}}^{\mathrm{H}}\mathbf{E}_{\mathbf{q}}, \tag{32}$$

where

$$\mathbf{G}_{\mathbf{q}} = \begin{pmatrix} \mathbf{J}_{\mathbf{q}} \\ (\mathbf{J}_{\mathbf{q}^i})^i \\ (\mathbf{J}_{\mathbf{q}^j})^j \\ (\mathbf{J}_{\mathbf{q}^k})^k \end{pmatrix}, \quad \mathbf{E}_{\mathbf{q}} = \begin{pmatrix} \mathbf{e} \\ \mathbf{e}^i \\ \mathbf{e}^j \\ \mathbf{e}^k \end{pmatrix}. \tag{33}$$

Lemma 2. *Suppose that* $\mathbf{g} : \mathbb{H}^N \to \mathbb{H}^M$ *has the GHR derivatives in the set* $S \subseteq \mathbb{H}^N$ *and that the GHR derivatives of* $\mathbf{g}$ *are Lipschitz continuous in* S *with the Lipschitz constant* L*, then for any* $\mathbf{q}, \mathbf{q}+\Delta\mathbf{q} \in S$

$$\left\|\mathbf{g}(\mathbf{q}+\Delta\mathbf{q}) - \mathbf{g}(\mathbf{q}) - \sum_{\mu\in\{1,i,j,k\}} \mathbf{J}_{\mathbf{q}^\mu}\Delta\mathbf{q}^\mu\right\| \le 2L\|\Delta\mathbf{q}\|^2. \tag{34}$$

Proof. Put $\mathbf{h}(t) = \mathbf{g}(\mathbf{q}+t\Delta\mathbf{q})$, where $0 \le t \le 1$. Using the chain rule in (1), the GHR derivative of $\mathbf{h}(t)$ can be found as

$$\mathbf{h}'(t) = \sum_{\mu\in\{1,i,j,k\}} \frac{\partial\mathbf{g}(\mathbf{q}+t\Delta\mathbf{q})}{\partial\mathbf{q}^\mu}(\Delta\mathbf{q})^\mu. \tag{35}$$

By substituting (35) into $\mathbf{h}(1) - \mathbf{h}(0) = \int_0^1 \mathbf{h}'(t)dt$ with $\mathbf{h}(0) = \mathbf{g}(\mathbf{q})$ and $\mathbf{h}(1) = \mathbf{g}(\mathbf{q}+\Delta\mathbf{q})$, we have

$$\begin{aligned} &\left\|\mathbf{g}(\mathbf{q}+\Delta\mathbf{q}) - \mathbf{g}(\mathbf{q}) - \sum_{\mu\in\{1,i,j,k\}} \mathbf{J}_{\mathbf{q}^\mu}\Delta\mathbf{q}^\mu\right\| \\ &= \left\|\int_0^1 \Big(\mathbf{h}'(t) - \sum_{\mu\in\{1,i,j,k\}} \mathbf{J}_{\mathbf{q}^\mu}\Delta\mathbf{q}^\mu\Big)dt\right\| \\ &= \left\|\int_0^1 \sum_{\mu\in\{1,i,j,k\}} \left(\frac{\partial\mathbf{g}(\mathbf{q}+t\Delta\mathbf{q})}{\partial\mathbf{q}^\mu} - \frac{\partial\mathbf{g}(\mathbf{q})}{\partial\mathbf{q}^\mu}\right)(\Delta\mathbf{q})^\mu dt\right\|. \end{aligned} \tag{36}$$

According to the Lipschitz condition of the GHR derivatives of $\mathbf{g}$, we arrive at

$$\left\|\mathbf{g}(\mathbf{q}+\Delta\mathbf{q}) - \mathbf{g}(\mathbf{q}) - \sum_{\mu\in\{1,i,j,k\}} \mathbf{J}_{\mathbf{q}^\mu}\Delta\mathbf{q}^\mu\right\| \le 4L\|\Delta\mathbf{q}\|^2 \int_0^1 tdt = 2L\|\Delta\mathbf{q}\|^2. \tag{37}$$

□

Theorem 3. *Let* $\mathbf{e} : \mathbb{H}^N \to \mathbb{H}^M$*, and let* $\ell = \mathbf{e}^{\mathrm{H}}\mathbf{e}$ *be twice real differential in the set* $S \subset \mathbb{H}^N$*. Assume that* $\mathbf{G}_{\mathbf{q}}, \mathbf{E}_{\mathbf{q}}$ *are Lipschitz continuous on* S *with constant* L*, that* $\| \mathbf{G}_{\mathbf{q}} \| \le \alpha$ *for all* $\mathbf{q} \in S$*, and that there exists* $\mathbf{q}_* \in D$ *such that* $\mathbf{G}_{\mathbf{q}_*}^{\mathrm{H}}\mathbf{E}_{\mathbf{q}_*} = 0$*,*

and λ is the smallest eigenvalue of $\mathbf{G}_{\mathbf{q}_*}^{\mathrm{H}}\mathbf{G}_{\mathbf{q}_*}$. If β defined in (45) satisfies $\beta < \lambda$, then for any $c \in (1, \lambda/\beta)$, there exists $\varepsilon > 0$ such that for all $\mathbf{q}_0 \in U(\mathbf{q}_*, \varepsilon)$, the sequence generated by the approximated Gauss-Newton method

$$\mathbf{q}_{n+1} = \mathbf{q}_n + (\mathbf{G}_{\mathbf{q}_n}^{\mathrm{H}}\mathbf{G}_{\mathbf{q}_n})^{-1}\mathbf{G}_{\mathbf{q}_n}^{\mathrm{H}}\mathbf{E}_{\mathbf{q}_n} \tag{38}$$

converges to $\mathbf{q}_*$, and obeys

$$\|\mathbf{q}_{n+1} - \mathbf{q}_*\|_2 \leq \frac{\lambda + c\beta}{2\lambda}\|\mathbf{q}_n - \mathbf{q}_*\|_2 < \|\mathbf{q}_n - \mathbf{q}_*\|_2. \tag{39}$$

Proof. For the sake of brevity, $\mathbf{G}_{\mathbf{q}_n}, \mathbf{E}_{\mathbf{q}_n}$ and $\mathbf{E}_{\mathbf{q}_*}$ are abbreviated $\mathbf{G}_n, \mathbf{E}_n$ and $\mathbf{E}_*$, respectively. Then the Gauss-Newton method (38) can be rewritten as

$$\begin{aligned}\mathbf{q}_{n+1} - \mathbf{q}_* &= \mathbf{q}_n - \mathbf{q}_* + (\mathbf{G}_n^{\mathrm{H}}\mathbf{G}_n)^{-1}\mathbf{G}_n^{\mathrm{H}}\mathbf{E}_n \\ &= (\mathbf{G}_n^{\mathrm{H}}\mathbf{G}_n)^{-1}\left(\mathbf{G}_n^{\mathrm{H}}\mathbf{E}_n + \mathbf{G}_n^{\mathrm{H}}\mathbf{G}_n(\mathbf{q}_n - \mathbf{q}_*)\right) \\ &= (\mathbf{G}_n^{\mathrm{H}}\mathbf{G}_n)^{-1}\left\{\mathbf{G}_n^{\mathrm{H}}\mathbf{E}_* + \mathbf{G}_n^{\mathrm{H}}\left(\mathbf{E}_n - \mathbf{E}_* + \mathbf{G}_n(\mathbf{q}_n - \mathbf{q}_*)\right)\right\}.\end{aligned} \tag{40}$$

By a familiar argument, there exists ε_1 such that $\mathbf{G}_n^{\mathrm{H}}\mathbf{G}_n$ is nonsingular and

$$\|(\mathbf{G}_n^{\mathrm{H}}\mathbf{G}_n)^{-1}\| \leq \frac{c}{\lambda} \text{ for } \mathbf{q}_n \in U(\mathbf{q}_*, \varepsilon_1), \tag{41}$$

where $c > 1$ and $\varepsilon_1 > 0$. Recalling that $\mathbf{G}_{\mathbf{q}}$ and $\mathbf{E}_{\mathbf{q}}$ in (33), then the Gauss-Newton method (40) can be expanded as

$$\begin{aligned}\mathbf{q}_{n+1} - \mathbf{q}_* = (\mathbf{G}_n^{\mathrm{H}}\mathbf{G}_n)^{-1}\big\{&\mathbf{G}_n^{\mathrm{H}}\mathbf{E}_* + \mathbf{J}_{\mathbf{q}_n}^{\mathrm{H}}(\mathbf{e}_n - \mathbf{e}_* + \mathbf{J}_{\mathbf{q}_n}(\mathbf{q}_n - \mathbf{q}_*)) \\ &+ (\mathbf{J}_{\mathbf{q}_n^i}^{\mathrm{H}})^i(\mathbf{e}_n - \mathbf{e}_*)^i + (\mathbf{J}_{\mathbf{q}_n^j}^{\mathrm{H}})^j(\mathbf{e}_n - \mathbf{e}_*)^j + (\mathbf{J}_{\mathbf{q}_n^k}^{\mathrm{H}})^k(\mathbf{e}_n - \mathbf{e}_*)^k \\ &+ (\mathbf{J}_{\mathbf{q}_n^i}^{\mathrm{H}}\mathbf{J}_{\mathbf{q}_n^i})^i(\mathbf{q}_n - \mathbf{q}_*)^i + (\mathbf{J}_{\mathbf{q}_n^j}^{\mathrm{H}}\mathbf{J}_{\mathbf{q}_n^j})^j(\mathbf{q}_n - \mathbf{q}_*)^j + (\mathbf{J}_{\mathbf{q}_n^k}^{\mathrm{H}}\mathbf{J}_{\mathbf{q}_n^k})^i(\mathbf{q}_n - \mathbf{q}_*)^k\big\}.\end{aligned} \tag{42}$$

From Lipschitz condition of $\mathbf{G}_{\mathbf{q}}$ and $\mathbf{G}_{\mathbf{q}_*}^{\mathrm{H}}\mathbf{E}_{\mathbf{q}_*} = 0$, we have

$$\|\mathbf{G}_{\mathbf{q}}^{\mathrm{H}}\mathbf{E}_*\| = \|(\mathbf{G}_{\mathbf{q}} - \mathbf{G}_{\mathbf{q}_*})^{\mathrm{H}}\mathbf{E}_*\| \leq \sigma\|\mathbf{q} - \mathbf{q}_*\|. \tag{43}$$

Recalling that $\mathbf{e} = \mathbf{d} - \mathbf{g}$ and using Lemma 2, we can estimate the second term in (42) as follows

$$\begin{aligned}\|\mathbf{e}_n - \mathbf{e}_* + \mathbf{J}_{\mathbf{q}_n}(\mathbf{q}_n - \mathbf{q}_*)\| &= \|\mathbf{g}(\mathbf{q}_n) - \mathbf{g}(\mathbf{q}_*) - \mathbf{J}_{\mathbf{q}_n}(\mathbf{q}_n - \mathbf{q}_*)\| \\ &\leq 2L\|\mathbf{q}_n - \mathbf{q}_*\|^2 + 3\alpha\|\mathbf{q}_n - \mathbf{q}_*\|.\end{aligned} \tag{44}$$

Combining (40)–(44) and $\|\mathbf{G}_n\| \leq \alpha$, yields

$$\begin{aligned}\|\mathbf{q}_{n+1} - \mathbf{q}_*\| &\leq \|(\mathbf{G}_n^{\mathrm{H}}\mathbf{G}_n)^{-1}\|\big\{\|\mathbf{G}_n^{\mathrm{H}}\mathbf{E}_*\| + \|\mathbf{J}_{\mathbf{q}_n}\|\|\mathbf{e}_n - \mathbf{e}_* + \mathbf{J}_{\mathbf{q}_n}(\mathbf{q}_n - \mathbf{q}_*)\| \\ &\quad + \|\mathbf{J}_{\mathbf{q}_n^i}\|\|\mathbf{e}_n - \mathbf{e}_*\| + \|\mathbf{J}_{\mathbf{q}_n^j}\|\|\mathbf{e}_n - \mathbf{e}_*\| + \|\mathbf{J}_{\mathbf{q}_n^k}\|\|\mathbf{e}_n - \mathbf{e}_*\| \\ &\quad + \|\mathbf{J}_{\mathbf{q}_n^i}^{\mathrm{H}}\mathbf{J}_{\mathbf{q}_n^i}\|\|\mathbf{q}_n - \mathbf{q}_*\| + \|\mathbf{J}_{\mathbf{q}_n^j}^{\mathrm{H}}\mathbf{J}_{\mathbf{q}_n^j}\|\|\mathbf{q}_n - \mathbf{q}_*\| + \|\mathbf{J}_{\mathbf{q}_n^k}^{\mathrm{H}}\mathbf{J}_{\mathbf{q}_n^k}\|\|\mathbf{q}_n - \mathbf{q}_*\|\big\} \\ &\leq \frac{c}{\lambda}\big\{\sigma\|\mathbf{q}_n - \mathbf{q}_*\| + 2L\alpha\|\mathbf{q}_n - \mathbf{q}_*\|^2 \\ &\quad + 3\alpha^2\|\mathbf{q}_n - \mathbf{q}_*\| + (3\alpha L + 3\alpha^2)\|\mathbf{q}_n - \mathbf{q}_*\|\big\} \\ &= \frac{c}{\lambda}\big\{\beta\|\mathbf{q}_n - \mathbf{q}_*\| + 2L\alpha\|\mathbf{q}_n - \mathbf{q}_*\|^2\big\},\end{aligned} \tag{45}$$

where $\beta = \sigma + 6\alpha^2 + 3\alpha L$. Thus, if $\beta < \lambda$, for any $c \in (1, \lambda/\beta)$, there exists $\varepsilon = \min\{\varepsilon_1, \frac{\lambda - c\beta}{4c\alpha L}\}$ such that for all $\mathbf{q}_n \in U(\mathbf{q}_*, \varepsilon)$, then

$$\begin{aligned} \|\mathbf{q}_{n+1} - \mathbf{q}_*\| &\leq \|\mathbf{q}_n - \mathbf{q}_*\| \left\{ \frac{c\beta}{\lambda} + \frac{2c\alpha L}{\lambda} \|\mathbf{q}_n - \mathbf{q}_*\| \right\} \\ &\leq \|\mathbf{q}_n - \mathbf{q}_*\| \left\{ \frac{c\beta}{\lambda} + \frac{\lambda - c\beta}{2\lambda} \right\} \\ &= \frac{\lambda + c\beta}{2\lambda} \|\mathbf{q}_n - \mathbf{q}_*\| < \|\mathbf{q}_n - \mathbf{q}_*\|. \end{aligned} \tag{46}$$

This shows that $\mathbf{q}_n$ converges to $\mathbf{q}_*$, the theorem follows. □

3.3 Quaternion Levenberg-Marquardt (L-M) algorithm

The L-M algorithm can be seen as a blend of vanilla gradient descent and Gauss-Newton iteration. It outperforms simple gradient descent and other conjugate gradient methods in a wide variety of problems

$$\begin{pmatrix} \Delta\mathbf{q} \\ \Delta\mathbf{q}^i \\ \Delta\mathbf{q}^j \\ \Delta\mathbf{q}^k \end{pmatrix} = (\mathbf{G}^{\mathrm{H}}\mathbf{G} + b\mathbf{I})^{-1}\mathbf{G}^{\mathrm{H}} \begin{pmatrix} \mathbf{e} \\ \mathbf{e}^i \\ \mathbf{e}^j \\ \mathbf{e}^k \end{pmatrix}, \tag{47}$$

where small values of b result in a Gauss-Newton update and large values of b result in a gradient descent update. Similar to the derivation of the Gauss-Newton algorithm, we obtain the following quaternion L-M update rule

$$\begin{aligned} \Delta\mathbf{q} = &\left(((\mathbf{H}_{11} + b\mathbf{I})^{-1} + \mathbf{L}\mathbf{T}^{-1}\mathbf{U})\mathbf{Q}_{11}^{\mathrm{H}} + \mathbf{L}\mathbf{T}^{-1}\mathbf{Q}_{12}^{\mathrm{H}} \right) \mathbf{e} \\ &+ \left(((\mathbf{H}_{11} + b\mathbf{I})^{-1} + \mathbf{L}\mathbf{T}^{-1}\mathbf{U})\mathbf{Q}_{21}^{\mathrm{H}} + \mathbf{L}\mathbf{T}^{-1}\mathbf{Q}_{22}^{\mathrm{H}} \right) \begin{pmatrix} \mathbf{e}^i \\ \mathbf{e}^j \\ \mathbf{e}^k \end{pmatrix}, \end{aligned}$$

and the approximated quaternion L-M update rule

$$\begin{aligned} \Delta\mathbf{q} &\approx (\mathbf{H}_{\mathbf{11}} + b\mathbf{I})^{-1} \left(\mathbf{J}_{\mathbf{q}}^{\mathrm{H}}\mathbf{e} + (\mathbf{J}_{\mathbf{q}^i}^{\mathrm{H}}\mathbf{e})^i + (\mathbf{J}_{\mathbf{q}^j}^{\mathrm{H}}\mathbf{e})^j + (\mathbf{J}_{\mathbf{q}^k}^{\mathrm{H}}\mathbf{e})^k \right) \\ &= (\mathbf{G}_{\mathbf{q}}^{\mathrm{H}}\mathbf{G}_{\mathbf{q}} + b\mathbf{I})^{-1}\mathbf{G}_{\mathbf{q}}^{\mathrm{H}}\mathbf{E}_{\mathbf{q}}. \end{aligned} \tag{48}$$

Theorem 4. *Let the conditions of Theorem 3 be satisfied. If $\beta < \lambda$, then for any $c \in (1, (\lambda + b)/(\beta + b))$, there exists $\varepsilon > 0$ such that for all $\mathbf{q}_k \in N(\mathbf{q}_*, \varepsilon)$ the sequence generated by the approxiated L-M method*

$$\mathbf{q}_{n+1} = \mathbf{q}_n + (\mathbf{G}_{\mathbf{q}_n}^{\mathrm{H}}\mathbf{G}_{\mathbf{q}_n} + b\mathbf{I})^{-1}\mathbf{G}_{\mathbf{q}_n}^{\mathrm{H}}\mathbf{E}_{\mathbf{q}_n} \tag{49}$$

converges to $\mathbf{q}_*$, and obeys to

$$\|\mathbf{q}_{n+1} - \mathbf{q}_*\|_2 \leq \frac{(\lambda + b) + c(\beta + b)}{2(\lambda + b)} \|\mathbf{q}_n - \mathbf{q}_*\|_2 < \|\mathbf{q}_n - \mathbf{q}_*\|_2. \tag{50}$$

Proof. This proof is similar to that of Theorem 3, so it is omitted here. □

4. Simulations

Simulations were performed in an M-step prediction setting and provide a comprehensive comparison between quaternion backpropagation (QBP) [1, 3], geometrical quaternion backpropagation (GQBP) [9] and quaternion Levenberg-Marquardt (QLM) for training a feedforward QNN. The feedforward QNN had one hidden layer comprising L inputs, eleven hidden neurons and one output neuron. In the experiments, the amplitudes of input signals in each dimension were scaled to within the range $[-0.8, 0.8]$ and the learning rate was chosen to be 0.1 for all algorithms. The quantitative performance measure was the prediction gain R_p, defined as [10]

$$R_p = 10 \log_{10} \frac{\sigma_x^2}{\sigma_e^2}, \tag{51}$$

where σ_x^2 and σ_e^2 denote the estimated variance of the input and the prediction error respectively. The considered quaternion-valued processes was the synthetic benchmark 4D Saito's Chaotic Signal [12]

$$\begin{bmatrix} \frac{\partial x_1}{\partial \tau} \\ \frac{\partial y_1}{\partial \tau} \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ -\alpha_1 & \alpha_1 \beta_1 \end{bmatrix} \begin{bmatrix} x_1 - \eta \rho_1 h(z) \\ y_1 - \eta \frac{\rho_1}{\beta_1} h(z) \end{bmatrix}, \tag{52}$$

$$\begin{bmatrix} \frac{\partial x_2}{\partial \tau} \\ \frac{\partial y_2}{\partial \tau} \end{bmatrix} = \begin{bmatrix} -1 & 1 \\ -\alpha_2 & \alpha_2 \beta_2 \end{bmatrix} \begin{bmatrix} x_2 - \eta \rho_2 h(z) \\ y_2 - \eta \frac{\rho_2}{\beta_2} h(z) \end{bmatrix}, \tag{53}$$

where $h(z)$ is the normalized hysteresis value given by

$$h(z) = \begin{cases} 1, & z \geqslant -1 \\ -1, & z \leqslant 1 \end{cases}, \tag{54}$$

and $z = x_1 + x_2$, $\rho_1 = \frac{\beta_1}{1-\beta_1}$, $\rho_2 = \frac{\beta_2}{1-\beta_2}$. The Saito chaotic signal was initialized with the following parameters: $\eta = 1.3$, $\alpha_1 = 7.5$, $\alpha_2 = 15$, $\beta_1 = 0.16$ and $\beta_2 = 0.097$. In Fig. 1, we depict the performance surface of the algorithms considered as a function of the prediction horizon M and the filter length L. We can see that the QLM outperformed the other two algorithms in all the cases, thus highlighting the advantages of the GHR calculus.

5. Conclusions

The quaternion gradient descent, quaternion Gauss-Newton and quaternion L-M algorithms have been developed for training feedforward QNN based on the GHR calculus, which greatly simplifies the working in the quaternion field. A widely linear quaternion least squares problem has been solved and utilized in the derivation of the algorithms. An approximated quaternion Gauss-Newton algorithm has been proposed to reduce the computational cost, which can be applied in general quaternion optimization problems. Under Lipschitz condition of the GHR derivatives, we have proved the convergence of the proposed quaternion learning algorithms.

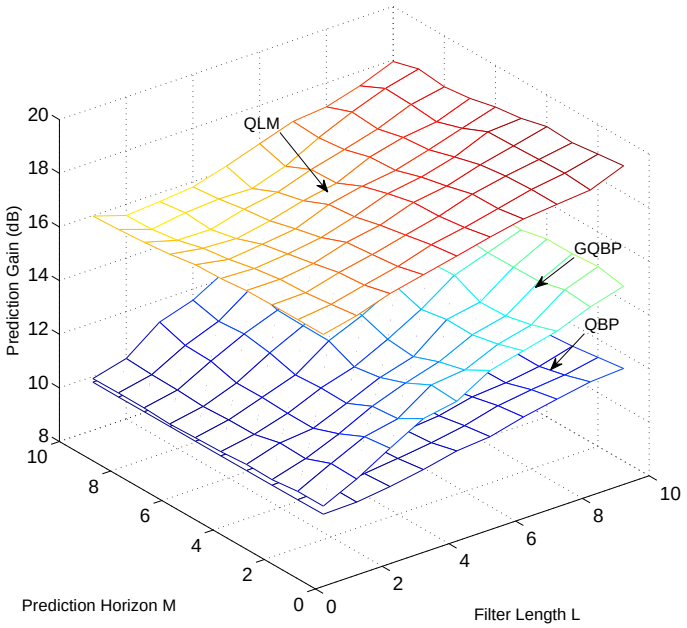

Fig. 1 *Performance of QLM, GQBP and QBP on the prediction of 4D Saito's chaotic signal.*

Acknowledgement

This work was supported by Foundation of Jilin Provincial Education Committee (No. JJKH20170914KJ), by Science and Technology Development Planning of Jilin Province (No. 20170520061JH), by the Fundamental Research Funds for the Central Universities (No. 2412016KJ014), and by the National Natural Science Foundation of China (Nos. 61301202, 61671099).

References

[1] ARENA P., FORTUNA L., MUSCATO G., XIBILIA M.G. Multilayer perceptrons to approximate quaternion valued functions. *Neural Netw.* 1997, 10(2), pp. 335–342, doi: 10.1016/s0893-6080(96)00048-2.

[2] BRANDWOOD D. A complex gradient operator and its application in adaptive array theory. *IEEE Commun. Radar Signal Process.* 130(1), pp. 1983:11–16, doi: 10.1049/ip-f-1.1983.0003.

[3] BUCHHOLZ S., BIHAN N.L. Polarized signal classification by complex and quaternionic multi-layer perceptrons. *Int. J. Neural Syst.* 2008, 18(2), pp. 75–85, doi: 10.1142/S0129065708001403.

[4] DEAVOURS C.A. The quaternion calculus. *Amer. Math. Monthly.* 1979, 80, pp. 995–1008.

[5] ELL T.A., SANGWINE S.J. Quaternion involutions and anti-involutions. *Comput Math. Applicat.* 2007, 53(1), pp. 137–143, doi: 10.1016/j.camwa.2006.10.029.

[6] JAHANCHAHI C., MANDIC D.P. A class of quaternion Kalman filters. *IEEE Trans. Neural Netw. Learn. Syst.* 2014, 25(3), pp. 533–544, doi: 10.1109/tnnls.2013.2277540.

[7] KREUTZ-DELGADO K. The complex gradient operator and the CR calculus [online]. ArXiv preprint, 2009 [viewed 2016-03-03]. Research report UCSD-ECE275CG-S2009v1.0. Available from: http://arxiv.org/abs/0906.4835.

[8] LEO S.D., ROTELLI P.P. Quaternionic analyticity. *App. Math. Lett.* 2003, 16(7), pp. 1077–1081, doi: 10.1016/S0893-9659(03)90097-8.

[9] MATSUI N., ISOKAWA T., KUSAMICHI H., PEPER F., NISHIMURA H. Quaternion neural network with geometrical operators. *J. Intell. Fuzzy Syst.* 2004, 15(3), pp. 149–164.

[10] MANDIC D.P., CHAMBERS J.A. *Recurrent Neural Networks for Prediction: Learning Algorithms, Architectures and Stability*. New York: Wiley, 2001, doi: 10.1002/047084535X.

[11] MANDIC D.P., JAHANCHAHI C., TOOK C.C. A quaternion gradient operator and its applications. *IEEE Signal Proc. Lett.* 2011, 18(1), pp. 47–50, doi: 10.1109/LSP.2010.2091126.

[12] MITSUBORI K., SAITO T. Torus doubling and hyperchaos in a five dimensional hysteresis circuit. In: *Proc. IEEE Int. Symp. Circuit Syst. (ISCAS 1994)*, London, U.K., IEEE, 1994, pp. 113–116.

[13] SUDBERY A. Quaternionic analysis. *Math. Proc. Camb Phil Soc.* 1979, 2, pp. 199–225, doi: 10.1017/S030500-4100055638.

[14] TOOK C.C., MANDIC D.P. The quaternion LMS algorithm for adaptive filtering of hypercomplex processes. *IEEE Trans. Signal Process.* 2009, 57(4), pp. 1316–1327, doi: 10.1109/TSP.2008.2010600.

[15] TOOK C.C., MANDIC D.P. Augmented second order statistics of quaternion random signals. *Signal Process.* 2011, 91(2), pp. 214–224, doi: 10.1016/j.sigpro.2010.06.024.

[16] UJANG B.C., TOOK C.C., MANDIC D.P. Split quaternion nonlinear adaptive filtering. *Neural Netw.* 2010, 23(3), pp. 426–434, doi: 10.1016/j.neunet.2009.10.006.

[17] VÍA J., RAMÍREZ D., SANTAMARíA I. Properness and widely linear processing of quaternion random vectors. *IEEE Trans. Inf. Theory.* 2010, 56(7), pp. 3502–3515, doi: 10.1109/TIT.2010.2048440.

[18] WIRTINGER W. Zur formalen theorie der funktionen von mehr komplexen veränderlichen. *Mathematische Annalen.* 1927, 97, pp. 357–375, doi: 10.1007/BF01447872.

[19] XIA Y., JAHANCHAHI C., MANDIC D.P. Quaternion-valued echo state networks. *IEEE Trans. Neural Netw. Learn. Syst.* 2015, 26(4), pp. 663–673, doi: 10.1109/TNNLS.2014.2320715.

[20] XIA Y., JAHANCHAHI C., NITTA T., MANDIC D.P. Performance bounds of quaternion estimators. *IEEE Trans. Neural Netw. Learn. Syst.* 2015, 26(12), pp. 3287–3292, doi: 10.1109/TNNLS.2015.2388782.

[21] XU D., JAHANCHAHI C., TOOK C.C., MANDIC D.P. Enabling quaternion derivatives: the generalized HR calculus. *R. Soc. Open Sci.* 2015, 2, 150255, doi: 10.1098/rsos.150255.

[22] XU D., Gao H., MANDIC D.P. A new proof of the generalized Hamiltonian–Real calculus. *R. Soc. Open Sci.* 2016, 3, 160211, doi: 10.1098/rsos.160211.

[23] XU D., MANDIC D.P. The theory of quaternion matrix derivatives. *IEEE Trans. Signal Process.* 2015, 63(6), pp. 1543–1556, doi: 10.1109/TSP.2015.2399865.

[24] XU D., XIA Y., MANDIC D.P. Optimization in quaternion dynamic systems: gradient, hessian, and learning algorithms. *IEEE Trans. Neural Netw. Learn. Syst.* 2016, 27(2), pp. 249–261, doi: 10.1109/TNNLS.2015.2440473.

[25] ZHANG F. *The Schur Complement and Its Applications*. Kluwer, Dordrecht: Springer, 2005, doi: 10.1007/b105056.