A Novel Continuous-Valued Quaternionic Hopfield Neural Network

Marcos Eduardo Valle

Department of Applied Mathematics
Institute of Mathematics, Statistics, and Scientific Computing
University of Campinas - Brazil

October 21, 2014

Introduction

- The Hopfield neural network (HNN) can be used for the storage and recall of *n*-bit vectors as an **associative memory**.
- Hyper-complex neural networks treat multi-dimensional data as a single entity.
  - Examples of hyper-complex models include complex-valued and quaternionic networks.
- In this talk, we introduce a kind of Hofield network for quaternions called **continuous-valued quaternionic Hopfield neural network** (CV-QHNN).
  - We investigate its convergence.
  - We evaluate its performance as an associative memory.

Basic Concepts on Quaternions

Quaternions have been introduced by Hamilton in 19th century as an extension of real and complex-numbers.

Definition

Quaternions A quaternion $q \in \mathbb{H}$ can be written as follows

$$q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}, \tag{1}$$

where $\mathbf{i}, \mathbf{j}$, and $\mathbf{k}$ are imaginary numbers that satisfy

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1. \tag{2}$$

Conjugate and norm

The conjugate $\bar{q}$ and the norm $|q|$ of a quaternion q are defined by

$$\bar{q} = q_0 - q_1\mathbf{i} - q_2\mathbf{j} - q_3\mathbf{k} \quad \text{and} \quad |q| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}. \tag{3}$$

Unit quaternions

A *unit* quaternion satisfies $|q| = 1$. The set of all unit quaternions is

$$\mathbb{S} = \{q \in \mathbb{H} : |q| = 1|\}. \tag{4}$$

Angle-phase representation

A quaternion q can be written as

$$q = |q|e^{\mathbf{i}\phi}e^{\mathbf{k}\psi}e^{\mathbf{j}\theta}, \tag{5}$$

where $\phi \in [-\pi, \pi)$, $\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right)$, and $\psi \in \left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$.

The exponentials are computed using the Euler's formula, for example,

$$e^{\mathbf{k}\psi} = \cos(\psi) + \mathbf{k}\sin(\psi). \tag{6}$$

Remark:

The conversion to angle-phase representation is not straightforward!

Hopfield Neural Networks

Given synaptic weights w_{ij} and a initial state

$$\mathbf{x}(0) = [x_1(0), \ldots, x_n(0)]^T \in [-1, +1]^n, \tag{7}$$

the (bipolar) Hopfield network defines asynchronously

$$x_i(t+1) = \begin{cases} \text{sgn}\big(v_i(t)\big), & v_i(t) \neq 0, \\ x_i(t), & v_i(t) = 0, \end{cases} \tag{8}$$

where $\text{sgn} : \mathbb{R} \setminus \{0\} \to \{-1, +1\}$ yields the signal of its argument and

$$v_i(t) = \sum_{j=1}^{n} w_{ij} x_j(t), \tag{9}$$

is the activation potential of the ith neuron at iteration t.

Remark:

The convergence is certain if $w_{ij} = w_{ji}$ and $w_{ii} \geq 0$.

Quaternionic Hopfield Neural Networks (QHNN

Given synaptic weights $w_{ij} \in \mathbb{H}$ and a quaternionic initial state

$$\mathbf{x}(0) = [x_1(0), \ldots, x_n(0)]^T, \tag{10}$$

the a QHNN defines asynchronously

$$x_i(t+1) = \begin{cases} f\big(v_i(t)\big), & v_i(t) \neq 0, \\ x_i(t), & v_i(t) = 0, \end{cases} \tag{11}$$

where f is a quaternion-valued activation function and

$$v_i(t) = \sum_{j=1}^{n} w_{ij} x_j(t), \tag{12}$$

is the activation potential of the ith neuron at iteration t.

Remark:

The main difference between QHNNs resides in the activation function!

QHNN as Associative Memories

- When we intent to use QHNN as an associative memory, we want to store a set of prototypes $\{\mathbf{u}^1, \ldots, \mathbf{u}^p\}$, called **fundamental memory set.**
- Subsequently, we expect to retrieve $\mathbf{u}^\xi$ even if the input represents an incomplete or distorted version $\tilde{\mathbf{u}}^\xi$ of $\mathbf{u}$

Formally, assuming that the sequence produced by a QHNN converges, we define

$$\mathcal{H}(\mathbf{x}) = \lim_{t \to \infty} \mathbf{x}(t), \quad \text{with } \mathbf{x}(0) = \mathbf{x}. \tag{13}$$

- The mapping $\mathcal{H}$ should satisfy $\mathcal{H}(\mathbf{u}^\xi) = \mathbf{u}^\xi$ for all $\xi = 1, \ldots, p$.
- Also, $\mathcal{H}$ is expect to exhibit some noise tolerance, i.e., $\mathcal{H}(\tilde{\mathbf{u}}^\xi) = \mathbf{u}^\xi$.

Remark:

One common problem associated with the design of AMs is the creation of spurious memories

Recording Recipes

In analogy to the real-valued Hopfield network, given a memory set $\{\mathbf{u}^1, \ldots, \mathbf{u}^p\}$, we usually define the synaptic weights according to:

Definition (Hebbian Rule or Correlation Recording Recipe)

$$w_{ij}^c = \frac{1}{n} \sum_{\xi=1}^{p} u_i^\xi \bar{u}_j^\xi, \quad \forall i, j \in \{1, 2, \ldots, n\}. \tag{14}$$

Hebbian rule is subject to the cross-talk between the prototypes!

Definition (Projection Rule or Generalized-inverse Recipe)

$$w_{ij}^p = \frac{1}{n} \sum_{\mu=1}^{p} \sum_{\nu=1}^{p} u_i^\xi (W_c)_{\mu\nu}^{-1} \bar{u}_j^\xi, \quad \forall i, j \in \{1, 2, \ldots, n\}, \tag{15}$$

where $(W_c)_{\mu\nu}^{-1}$ denotes μ, ν entry of the inverse of $W_c = (w_{ij}^c)$.

Proposition

The synaptic weights produced by either Hebbian rule or projection rule satisfy the conditions

$$w_{ij} = \bar{w}_{ji} \quad \text{and} \quad w_{ii} \geq 0, \quad \forall i,j \in \{1,2,\ldots,n\}. \tag{16}$$

Let us now turn our attention to two possible choices for the activation function f of a QHNN. Namely,

- Multivalued signum function.
- Continuous-valued signum function.

Remark:

We would like to recall that choosing an appropriate activation function is a challenging problem in quaternionic neural networks.

Multivalued Signum Function

The quaternionic multivalued signum function, introduced by Isokawa et al., generalizes the complex-valued signum function to quaternions.

Definition (Multivalued Signum Function)

Let $|q|e^{\mathbf{i}\phi}e^{\mathbf{k}\psi}e^{\mathbf{j}\theta}$ be the angle-phase representation of a quaternion $q \neq 0$. Then, $p = \mathrm{qsgn}(q)$ is the unit quaternion determined by

$$p = -e^{\phi_0 \left\lfloor \frac{\pi+\phi}{\phi_0} \right\rfloor \mathbf{i}} e^{\left(-\frac{\pi}{4}+\psi_0 \left\lfloor \frac{\frac{\pi}{4}+\psi}{\psi_0} \right\rfloor\right)\mathbf{k}} e^{\left(-\frac{\pi}{2}+\theta_0 \left\lfloor \frac{\frac{\pi}{2}+\theta}{\theta_0} \right\rfloor\right)\mathbf{j}}. \tag{17}$$

where

$$\phi_0 = 2\pi/K_1, \quad \psi_0 = \pi/(2K_2), \quad \text{and} \quad \theta_0 = \pi/K_3, \tag{18}$$

are the phase quantum defined by integers K_1, K_2, and K_3.

Loosely speaking, the qsgn function yields a representative point in a section of the hypersphere $\mathbb{S}$ of all unit quaternions.

Proposition (Isokawa et al.)

Let $\bar{w}_{ij} = w_{ji}$ and $w_{ii} \geq 0$ for any indexes i, j. A QHNN with qsgn always converge to a stationary state if the ***angle-phases*** *of a quaternionic neuron* ***are never updated simultaneously***.

In practice, the qsgn must be slightly modified in order to ensure the convergence of the QHNN to a stationary state!
Formally, if $v_i(t) \neq 0$, the ith neuron is updated according to

$$x_i(t+1) = \begin{cases} -e^{\Phi \mathbf{i}} e^{\left(-\frac{\pi}{4} + \psi_0 \left\lfloor \frac{\frac{\pi}{4} + \psi}{\psi_0} \right\rfloor\right)\mathbf{k}} e^{\left(-\frac{\pi}{2} + \theta_0 \left\lfloor \frac{\frac{\pi}{2} + \theta}{\theta_0} \right\rfloor\right)\mathbf{j}}, \\ \text{or} \\ -e^{\phi_0 \left\lfloor \frac{\pi + \phi}{\phi_0} \right\rfloor \mathbf{i}} e^{\left(-\frac{\pi}{4} + \psi_0 \left\lfloor \frac{\frac{\pi}{4} + \psi}{\psi_0} \right\rfloor\right)\mathbf{k}} e^{\Theta \mathbf{j}}, \end{cases} \tag{19}$$

where ϕ_0, ψ_0, θ_0 are given by (18) and Φ and Θ are angle-phases of the current state, that is, $x_i(t) = e^{\Phi \mathbf{i}} e^{\Psi \mathbf{k}} e^{\Theta \mathbf{j}}$.

Continuous-Valued Signum Function

Motivation:

Aizenberg et al. proposed a continuous-valued activation function which is defined by the limit of the complex-signum function as the number of sections tends to ∞.

Idea:

We define the quaternionic continuous-valued signum function σ by the limit of the of qsgn as $K_1, K_2, K_3 \to \infty$.

Definition (Continuous-Valued Signum Function)

The continuous-valued signum function is given by

$$\sigma(z) = \frac{z}{|z|}, \qquad \forall z \neq 0. \tag{20}$$

Note that σ does not depend on a certain representation of z.

The following theorem, whose proof can be found in the paper, provides a sufficient condition for the convergence of a sequence produced by a CV-QHNN.

Theorem

The sequence produced by a QHNN with σ in an asynchronous update mode is convergent for any initial state $\mathbf{x}(0) \in \mathbb{S}^n$ if the synaptic weights satisfy $w_{ij} = \bar{w}_{ji}$ and $w_{ii} > 0$ for any $i, j \in \{1, \ldots, n\}$.

Conclusion:

- CV-QHNN can be used to implement an AM if $w_{ij} = \bar{w}_{ji}$ and $w_{ii} \geq 0$ for any indexes i, j.
- In contrast to the MV-QHNN, all the components of the ith quaternionic neuron can be updated simultaneously.
- CV-QHNN is computationally cheaper than the MV-QHNN.

Computational Experiments

Let us evaluate the storage capacity and the noise tolerance of the CV-QHNN model synthesized using either Hebbian or projection rules.

We performed the following steps 100 times with $n = 100$ fixed and different values for p:

1. We randomly generated a fundamental memory set $\mathcal{U} = \{\mathbf{u}^1, \ldots, \mathbf{u}^p\} \subseteq \mathbb{S}^n$ in which the components u_i^ξ's are independent and uniformly distributed.
2. We also generated a vector $\mathbf{z} = [z_1, \ldots, z_n]^T \in \mathbb{S}^n$ whose components are independent and uniformly distributed.
3. We defined the input $\mathbf{x}^\alpha = [x_1^\alpha, \ldots, x_n^\alpha]^T \in \mathbb{S}^n$ as follows for all $\alpha \in \{0, 0.05, 0.1, 0.15, \ldots, 1\}$:

$$x_i^\alpha = \frac{(1-\alpha)u_i^1 + \alpha z_i}{|(1-\alpha)u_i^1 + \alpha z_i|}, \quad \forall i = 1, \ldots, n. \tag{21}$$

3 For each α, we computed the error rates:

a) The **input error** $D_\alpha = d_2(\mathbf{x}^\alpha, \mathbf{u}^1)$

b) The **output error** $E_\alpha^{c1} = d_2(\mathbf{x}^c(1), \mathbf{u}^1)$ obtained after a **single step** of the CV-QHNN based on the **correlation rule**.

c) The **output error** $E_\alpha^{c*} = d_2(\lim \mathbf{x}^c(t), \mathbf{u}^1)$ obtained **after convergence** of the CV-QHNN synthesize using the **correlation rule**.

d) The **output error** $E_\alpha^{p1} = d_2(\mathbf{x}^c(1), \mathbf{u}^1)$ obtained after a **single step** of the CV-QHNN based on the **projection rule**.

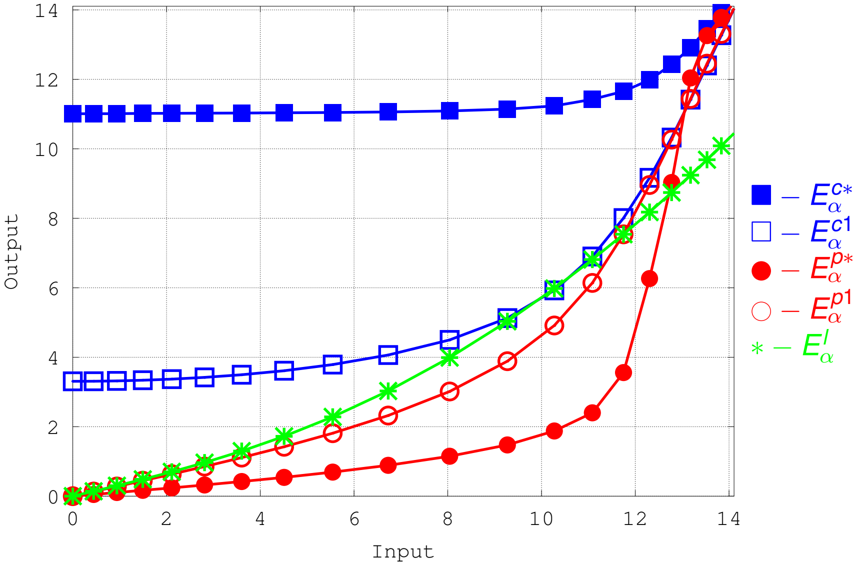
e) The **output error** $E_\alpha^{p*} = d_2(\lim \mathbf{x}^c(t), \mathbf{u}^1)$ obtained **after convergence** of the CV-QHNN synthesize using the **projection rule**.

f) The **output error** $E_\alpha^{l} = d_2(W^p \mathbf{x}^\alpha, \mathbf{u}^1)$ produced by a linear model with the matrix W^p determined using the **projection rule**.

Remark:

E_α^l corresponds to the error rate produced by a quaternion-valued version of the *optimal linear associative memory* (OLAM).

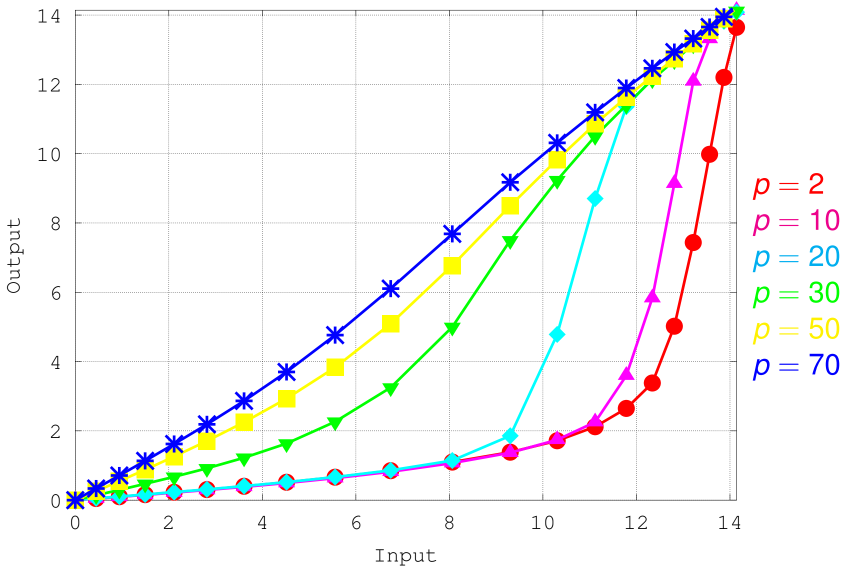
Average D_α by the average output error for $p = 10$.



Conclusion:

- $E^{c*}_{\alpha} > 0$ and $E^{c1}_{\alpha} > 0$ when $D_{\alpha} = 0$. Thus, the CV-QHNN synthesized using the Hebbian rule usually failed to recall an undistorted pattern.
- $E^{c*}_{\alpha} \geq E^{c1}_{\alpha}$. Thus, iterations of the CV-QHNN with the Hebbian rule introduces a noisy due to the cross-talk.
- $E^{p*}_{\alpha} = 0$, $E^{p1}_{\alpha} = 0$, and $E^{l}_{\alpha} = 0$ when $D_{\alpha} = 0$. Hence, the CV-QHNN with the projection rule and the quaternionic version of the OLAM yielded perfect recall of the original patterns.
- $E^{p*}_{\alpha} \leq E^{p1}_{\alpha}$ for $D_{\alpha} < 13$ shows that some amount of noise is removed of $\mathbf{x}^{p}(t)$ until a stationary state is reached.
- $E^{p1}_{\alpha} \leq E^{l}_{\alpha}$ for $D_{\alpha} < 11$, implies that the activation function σ improves the noise tolerance of the CV-QHNN.

Average D_α by the average E_α^{p*}.



Conclusion:

- The noise tolerance reduces as the number p of fundamental memories increase.
- The error rates produced by $p = 2$ and $p = 10$ are very similar for $D_\alpha < 11$.
- We may guess that the noise tolerance begins to deteriorate for p larger than $0.1n$.

Concluding Remarks

- We revised the class of *quaternionic Hopfield neural networks*.
- Particular attention was given to the *multivalued QHNN* (MV-QHNN) proposed by Isokawa et al.
- Motivated by the works of Aizenberg et al., we introduced the *continuous-valued quaternionic Hopfield neural network* (CV-QHNN).
- The CV-QHNN can be implemented and analyzed more easily than the MV-QHNN.
- In contrast to the MV-QHNN, all the components of the *i*th neuron can be updated simultaneously in the CV-QHNN.
- On the downside, the novel CV-QHNN may exhibit more spurious memories than the MV-QHNN model.

Concluding Remarks

- We also reviewed the Hebbian and projection rules used to synthesize a QHNN.
- The CV-QHNN synthesized using the Hebbian rule is subject to the cross-talk between the fundamental memories.
- The CV-QHNN with the projection rule exhibit optimal absolute storage capacity, that is, the original vectors are retrieved correctly under presentation of undistorted fundamental memories.
- Computational experiments revealed that the CV-QHNN with the projection rule exhibit some noise tolerance (which deteriorate by increasing the number of fundamental memories).

Future Researches

- We plan to investigate further the noise tolerance of the CV-QHNN with the projection rule.
- A study on the spurious memories of CV-QHNN models is also necessary.
- We intent to compare the performance of the novel CV-QHNN with the MV-QHNN model.
- We plan to investigate applications of the CV-QHNN, for instance, in the reconstruction of corrupted color images.

Thank you!

This work was supported in part by:

FAPESP CNPq FAEPEX