

## Convergence and Stability of Quantized Hopfield Networks Operating in a Fully Parallel Mode

**Daniel Calabuig**

*dacaso@iteam.upv.es*

**Jose F. Monserrat**

*jomondel@iteam.upv.es*

**Narcís Cardona**

*ncardona@iteam.upv.es*

*Universidad Politécnica de Valencia, Institute of Telecommunications and Multimedia Applications, Valencia 46022, Spain*

Hopfield neural networks (HNNs) have proven useful in solving optimization problems that require fast response times. However, the original analog model has an extremely high implementation complexity, making discrete implementations more suitable. Previous work has studied the convergence of discrete-time and quantized-neuron models but has limited the analysis to either two-state neurons or serial operation mode. Nevertheless, two-state neurons have poor performance, and serial operation modes lose fast convergence, which is characteristic of analog HNNs. This letter is the first in the field analyzing the convergence and stability of quantized Hopfield networks (QHNs)—with more than two states—operating in fully parallel mode. Moreover, this letter presents some further analysis on the energy minimization of this type of network. The main conclusion drawn is that QHNs operating in fully parallel mode always converge to a stable state or a cycle of length two and any stable state is a local minimum of the energy.

### 1 Introduction ---

The main feature of iterative algorithms is that new solutions are obtained from the result of a previous computation, and with each update, the solution approaches closer to the final optimum. This behavior is useful in minimization problems where algorithms are continuously approaching the minimum. Algorithms like Newton's or steepest descent (Moon & Stirling, 2000) have proven their good performance in finding the minimum of functions in  $\mathbb{R}^N$ .

Hopfield proposed the use of recurrent neural networks for solving optimization problems via the minimization of an energy function (Hopfield & Tank, 1985). From a hardware point of view, the Hopfield neural network (HNN) is an analog network composed of several operational amplifiers

interconnected by resistors (Hopfield, 1984). Each neuron output is bounded between a maximum and a minimum value, which reduces the search space to an  $N$ -dimensional hypercube. Neurons are considered active or inactive if their outputs exceed a threshold or not. The stability of this kind of network is guaranteed when the energy function is described as a Lyapunov function (Haykin, 1999).

Starting from Hopfield's work, HNNs have been successfully employed in several practical problems due to their fast convergence (see examples in Lázaro & Girma, 2000, and Ahn & Ramakrishna, 2004), although many more can be found in the literature—a direct consequence of neuron parallel interworking. In these applications, HNNs can replace other heuristics with worse performance whose use is motivated only by the need of a fast response time.

**1.1 Quantized Hopfield Networks.** The dynamics of the discrete-time and quantized-neuron model, also known as quantized Hopfield networks (QHNs), is, in general, different from that of the continuous-neuron model (Bousño-Calzón & Salcedo-Sanz, 2004). Although the original HNN is a hardware model using analog circuits, the enormous size of the hardware network with a large number of neurons and the difficulty of accurately implementing the resistor values, which can change network behavior, have made QHNs implemented over digital devices, like field-programmable gate arrays (FPGAs), the best HNN implementation option. Therefore, QHNs are used in computer simulations and, most of all, in realistic hardware implementations of HNNs, since quantization can be conveniently adjusted to the hardware limits of the digital device. When reducing the quantization step, QHNs tend to the digital implementation of continuous HNNs (CHNs). Section 2 presents a comparison of QHNs and CHNs. First, let us formally define QHNs.

Let  $Q$  be a QHN of  $N$  neurons uniquely defined by  $(\mathbf{T}, \mathbf{i})$  where  $\mathbf{T} = [T_{ij}]$  is an  $N \times N$  symmetric matrix and  $\mathbf{i} = [I_i]$  is a vector of  $N$  elements,<sup>1</sup> where  $N$  is the number of neurons. The network state at iteration  $t$  is defined by the neuron outputs  $\mathbf{v}(t) = [V_i(t)]$  that are updated by  $\Delta_i(t)$ , that is,  $V_i(t+1) = V_i(t) + \Delta_i(t)$ . Neuron outputs are quantized over the interval  $[0, 1]$  by steps of  $\Delta = 1/p$ ; thus,  $V_i(t) \in S$ , where  $S \equiv \{0, \Delta, 2\Delta, \dots, p\Delta\}$  is the set of all possible states of a neuron of  $Q$ . An energy function defined as

$$E(t) = -\frac{1}{2} \mathbf{v}^T(t) \mathbf{T} \mathbf{v}(t) - \mathbf{v}^T(t) \mathbf{i} \quad (1.1)$$

---

<sup>1</sup>Further details on the physical meaning of the HNN parameters can be found in Hopfield (1984).

is associated with each QHN  $Q$ . If neuron  $i$  is updated at iteration  $t$ , then

$$\Delta_i(t) = \begin{cases} -\Delta, & \frac{\partial E}{\partial V_i}(t) > 0, V_i(t) > 0, \\ \Delta, & \frac{\partial E}{\partial V_i}(t) < 0, V_i(t) < 1, \\ 0, & \text{otherwise,} \end{cases} \quad (1.2)$$

$$\frac{\partial E}{\partial V_i}(t) = - \sum_{j=1}^N T_{ij} V_j(t) - I_i, \quad (1.3)$$

and  $\Delta_i(t) = 0$  if the neuron is not updated. Depending on how neurons are updated, several modes of operation have been identified (Bruck & Goodman, 1988).  $Q$  operates in a serial mode if one neuron is updated at each iteration  $t$ , in a parallel mode if  $n < N$  neurons are updated, or in fully parallel mode if all  $N$  neurons are updated each iteration.

**1.2 State of the Art on Convergence and Stability of QHNs.** In Bruck and Goodman (1988), the behavior of binary Hopfield networks (BHNs) was studied: two-state neurons with  $p = 1$ , operating in serial and fully parallel modes. The main conclusions were that BHNs operating in serial mode always converge to a stable state if the elements of the diagonal of  $\mathbf{T}$  are nonnegative and that BHNs operating in fully parallel mode converge to a stable state or to a cycle of length two, that is, BHN oscillates between two states, independent of  $\mathbf{T}$ . Moreover, Bruck and Goodman demonstrated that the energy is a monotonically decreasing function in the serial-operation mode, obviously with a nonnegative diagonal in  $\mathbf{T}$ . These conclusions have important implications. First, the fact that BHN stability is guaranteed when operating in serial mode solves one of the main drawbacks of CHNs, which were criticized precisely because of their problems of instability (Wilson & Pawley, 1988; Forti, Manetti, & Marini, 1992). Second, the final conclusion of Bruck and Goodman entails that BHNs in serial-operation mode evolve toward a local minimum of the energy function. Therefore, they exhibit the same ability as CHNs for solving optimization problems with a cost function to be minimized, which is the end purpose of these mathematical tools. However, this conclusion is valid only for the serial-operation mode that entails a significant deterioration of the system response time, since only one neuron is updated per iteration.

Two-state neurons are a simplification of the continuous-neuron model and assume that neurons can take only two values in their evolution. Hopfield (1984) proposed these for solving convergence problems that he detected in his original model. Nevertheless, the use of BHNs has severe consequences, as shown in Calabuig, Monserrat, Gómez-Barquero, and Lázaro (2006), and their outcomes are very poor compared with CHNs (Joya, Atencia, & Sandoval, 2002). When CHNs and QHNs are compared,

the performance of a QHN depends on the shape of the energy function, the number of neurons, and the value of  $p$ . Obviously, with a greater  $p$ , QHNs are more similar to CHNs. For these reasons, QHNs with  $p > 1$  are preferable, although high values of  $p$  require greater response times.

The generalization of BHNs to QHNs was originally proposed by Matsuda (1999a), and many others have used them since then (Bousoño-Calzón & Salcedo-Sanz, 2004; Matsuda, 1999b). The QHNs proposed by Matsuda operate in serial mode. Considering this mode of operation, Matsuda (1999a) demonstrated the convergence of QHNs to a local minimum of the energy, which reinforces the same statement made in Bruck and Goodman (1988) for BHNs. Moreover, using a modified dynamics, not exactly the one shown in equation 1.2, the convergence to a minimum was ensured independent of the diagonal of  $\mathbf{T}$ .

**1.3 Objectives of This Work.** Although this letter does not focus on demonstrating the good performance of QHNs, since this issue has already been addressed in Calabuig et al. (2006) and Joya et al. (2002), this letter demonstrates, in section 2, the power of QHNs operating in fully parallel mode. Our main goal is to prove that this type of network can be implemented in a way such that it is fast and stable. With this aim, this letter extends the work of Bruck and Goodman (1988) and Matsuda (1999a) and analyzes the convergence and stability of QHNs with  $p > 1$  and fully parallel operation mode. Section 3 demonstrates that this type of network converges to a stable state or to a cycle of length two. Although the energy can increase from one iteration to the next, section 4 shows that if the network converges, it does so toward a local minimum.

## 2 Advantages of QHNs Operating in Fully Parallel Mode \_\_\_\_\_

One of the main drawbacks of the serial-operation mode is that it loses the fast convergence of HNNs, characterized by the parallel and simultaneous evolution of neurons, making serial operation useless for applications requiring fast response times.

Another drawback is that the convergence to a stable state is ensured only if the elements of the diagonal of  $\mathbf{T}$  are nonnegative, which is not always true. If this condition is not satisfied, then the QHN may converge to cycles of unknown length, which makes the detection and exit from these cycles practically impossible. Matsuda (1999a) tried to solve this by slightly modifying the updating criteria of equation 1.2. Nevertheless, this approach was never compared with a fully parallel operation mode.

Another solution is to force zeros in the diagonal. HNNs are usually used in optimization problems that can be described with a set of binary

variables.<sup>2</sup> In fact, the use of binary variables represented by neurons is a common practice in the application of HNNs to engineering problems (see the examples in Hopfield & Tank, 1985; Lázaro & Girma, 2000; Calabuig, Monserrat, Gómez-Barquero, & Cardona, 2008). Therefore, the desired solutions are at the hypercube corners, since only the corners have a physical meaning in the original problem. At the hypercube corners,  $V_i = 0$  or  $V_i = 1$  and, consequently,  $V_i^2 = V_i$ . Thus, the quadratic term  $T_{ii}$  can be integrated in the linear term  $I_i$  without changing the energy value at the corners. Moreover, the corner with minimum energy remains the same. Therefore, with a simple modification of the energy function, the elements of the diagonal of  $\mathbf{T}$  can be forced to be equal to zero. Nevertheless, this is not always a good option. Although the energy at the corners is not changed, the shape of the energy function inside the hypercube changes drastically and may produce incorrect outcomes.

This section compares these two serial approaches, Matsuda's proposal (SQHN-Mat) and zero forcing (SQHN), with the fully parallel operation mode (PQHN) and CHNs. For this analysis, the energy function defined in Le and Pham (2005) for the  $M$ -queens problem (MQP) was used (see the appendix for further details about the simulations). In this case, all the elements of the diagonal of  $\mathbf{T}$  are negative. All results depicted in Figures 1 and 2 are averaged over 1000 independent simulations with different initial states. Figure 1 shows the performance of the four approaches for an increasing number of queens. The CHN always has less energy than the other three approaches, as shown in Figure 1a. The other QHNs use 64 neuron states, that is,  $p = 63$ .

For the SQHN, the diagonal was cancelled to guarantee its convergence. In spite of having the same energy values at the corners as PQHN, the change of the energy shape affects drastically the convergence of SQHNs. Indeed, results prove that SQHN is far from the optimum.

Following from the previous example, the results of Figure 1b demonstrate that PQHNs need significantly fewer iterations to converge than the other three approaches. The time to converge is more than one order of magnitude below that of CHNs and more than two for many queens with respect to SQHN-Mat. Nevertheless, theorem 1 will prove that PQHNs may converge to a stable state or a cycle of length two, whereas the other three approaches always converge to a stable state. This convergence to cycles slightly increases the average energy with respect to CHN and SQHN-Mat, as shown in Figure 1a.

In order to have a measurement more suitable for comparing the performance of the four approaches, Figure 1c depicts the average number

---

<sup>2</sup>Although variables in the original problem are binary, neurons characterizing those variables should have more than two states to improve the outcomes.

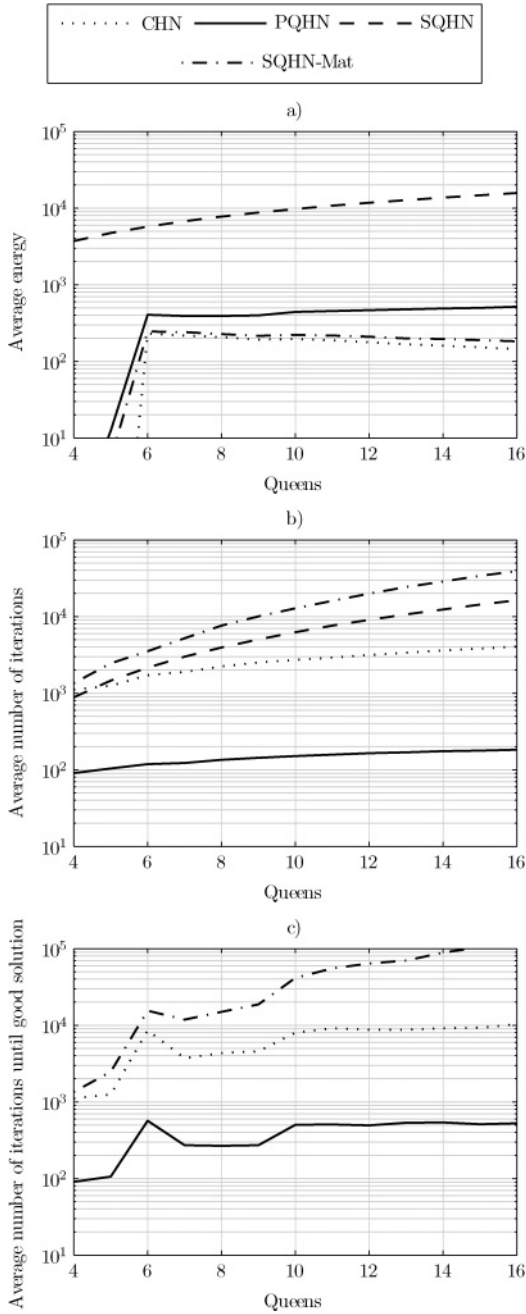


Figure 1: Performance for different numbers of queens.

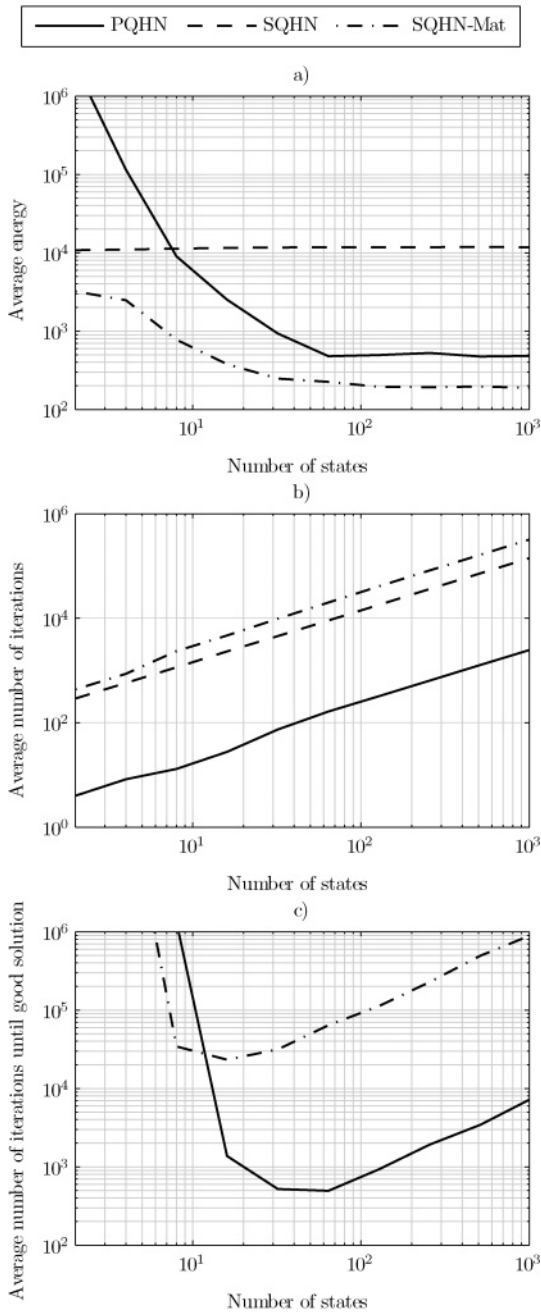


Figure 2: Performance for different numbers of neuron states.

of iterations until a good solution is reached or, in other words, until the network converges to a valid solution of the problem. After a network ends its evolution, on detecting a stable state or a cycle of length two, neuron outputs are rounded to the extremes, 0 or 1. If that state is not a valid solution of the MQP, the algorithms are run again with a different initial state. The results of Figure 1c show the total number of iterations needed to reach a good solution. This figure shows that the PQHN has the best behavior, improving the CHN and SQHN-Mat by a factor similar to that of Figure 1b. This means that despite the average energy being slightly higher, the final states of the PQHN, even after detecting a cycle,<sup>3</sup> are not far from a good solution, and rounding solves the problem in many cases. In fact, the probability of reaching valid solutions is very similar for the PQHN and SQHN-Mat, which is 45.7%, and 45.8%, respectively. CHNs reach good solutions 49.0% of the time, which is only slightly higher. Nevertheless, SQHNs could not converge to a valid solution in any of the trials. This is due to the change performed in the diagonal of  $\mathbf{T}$ .

Finally, the behavior of PQHNs relies completely on the number of states. Figure 2 compares different numbers of states for the 12-queens case. SQHN is barely affected by the number of states in terms of average energy—because of the modification of the diagonal of  $\mathbf{T}$ —whereas PQHN and SQHN-Mat require a certain number of states to show good performance. This figure also demonstrates why BHNs are not a good option, since a QHN's performance worsens for such a low number of states. The average number of iterations required to reach a good solution is also depicted in Figure 2c. This figure shows that at its best, SQHN-Mat needs many more iterations than PQHN.

To sum up, this section has demonstrated, with a simple example, the advantages of QHNs operating in fully parallel mode. They present good, fast convergence behavior compared with CHNs and SQHN-Mat and much better performance than SQHNs. The rest of the letter studies the convergence and stability of PQHNs.

### 3 Convergence and Stability

---

Bruck and Goodman (1988) proved that BHNs operating in a fully parallel mode converge to a stable state or a cycle of length two. The following theorem shows that any QHN with  $p > 1$  can be reformulated in the form of a BHN; hence, the same conclusion remains valid.

**Theorem 1.** *Let  $Q \equiv (\mathbf{T}, \mathbf{i})$  be a QHN with  $p > 1$  and  $N$  neurons and  $\mathbf{v}(t)$  the neuron outputs at iteration  $t$ . Let  $S \equiv \{s_n = n\Delta : n = 0, \dots, p\}$  and  $\hat{S} \equiv \{0, 1\}$ .*

---

<sup>3</sup>Note that PQHN may converge to cycles. In that case, the final state is one of the two states of the cycle.



Then there exists a BHN  $\hat{Q} \equiv (\hat{T}, \hat{\mathbf{i}})$  of  $\hat{N}$  neurons and an injective function  $F : S^N \rightarrow \hat{S}^{\hat{N}}$  so that  $F(\mathbf{v}(t)) = \hat{\mathbf{v}}(t) \forall t$  if  $F(\mathbf{v}(0)) = \hat{\mathbf{v}}(0)$ .

**Proof.** In order to prove the theorem, at least one specific BHN  $\hat{Q}$  and a function  $F$  satisfying the above conditions must be found.

First, consider the following transformation. Divide each neuron of  $Q$  into  $p$  ordered binary neurons in such a way that if the original neuron is in state  $s_n$ , the first  $n$  binary neurons are in state 1 and the rest in state 0. For example, if  $p = 4$ , the original neurons can be in any of the following states  $S \equiv \{0, 0.25, 0.5, 0.75, 1\}$ . Now each neuron is divided into four binary neurons, and their states are selected according to this rule:  $0 \leftrightarrow [0 \ 0 \ 0 \ 0]$ ,  $0.25 \leftrightarrow [1 \ 0 \ 0 \ 0]$ ,  $0.5 \leftrightarrow [1 \ 1 \ 0 \ 0]$ ,  $0.75 \leftrightarrow [1 \ 1 \ 1 \ 0]$ , and  $1 \leftrightarrow [1 \ 1 \ 1 \ 1]$ . Assume this transformation is the injective function  $F$ .

Since each neuron is divided into  $p$  binary neurons, the BHN  $\hat{Q}$  has  $\hat{N} = pN$  neurons. In order to simplify the nomenclature in this proof and as a direct consequence of this transformation, rewrite  $\hat{Q}$  as a 2D-BHN,  $Q^* = (\mathbf{T}^*, \mathbf{i}^*)$ , with  $N \times p$  neurons, where the  $p$  neurons of row  $i$  are related to the  $i$ th neuron of  $Q$ . It is worth noting that 2D-HNNs do not differ from HNNs; they are just a way to order the neurons and group them according to their specific meaning. Therefore, if the initial states are the same,  $F(\mathbf{v}(0)) = \mathbf{v}^*(0)$ , and the dynamics of  $Q$  and  $Q^*$  are equivalent, then  $F(\mathbf{v}(t)) = \mathbf{v}^*(t), \forall t$ . Thus, if at iteration  $t$ , the  $i$ th neuron of  $Q$  increases, then the first neuron at state 0 of the  $i$ th row of  $Q^*$  must change to state 1. Also, if the  $i$ th neuron of  $Q$  decreases, the last neuron at state 1 of the  $i$ th row of  $Q^*$  must change to state 0. Consider the following dynamics:

$$\frac{\partial E^*}{\partial V_{ij}^*}(t) = A(1 - V_{ij-1}^*(t) - V_{ij+1}^*(t)) + \frac{\partial E}{\partial V_i}(t), \tag{3.1}$$

$$V_{i0}^*(t) = 1, \quad V_{ip+1}^*(t) = 0, \tag{3.2}$$

$$A > \max_k \left( \sum_{l=1}^N |T_{kl}| + |I_k| \right), \tag{3.3}$$

where  $E^*$  is the energy function of  $Q^*$  and  $V_{ij}^*$  is the neuron in the  $i$ th row and  $j$ th column of  $Q^*$ . The neurons  $(i, 0)$  and  $(i, p + 1)$  do not exist, and consequently,  $V_{i0}^*(t)$  and  $V_{ip+1}^*(t)$  are defined in equation 3.2. Moreover,  $A$  is a positive constant greater than the absolute value of any  $\partial E / \partial V_i(t)$ . From equation 3.1, if both  $V_{ij-1}^*(t)$  and  $V_{ij+1}^*(t)$  are in state 1, the energy gradient of the neuron  $(i, j)$  is negative, and, according to equation 1.2,  $V_{ij}^*(t + 1) = 1$ . Similarly, if both  $V_{ij-1}^*(t)$  and  $V_{ij+1}^*(t)$  are in state 0, the energy gradient of the neuron  $(i, j)$  is positive, and in the next iteration,  $V_{ij}^*(t + 1) = 0$ . If  $V_{ij-1}^*(t) = 1$  and  $V_{ij+1}^*(t) = 0$ , then the energy gradient of the neuron  $(i, j)$  is exactly the energy gradient of the  $i$ th neuron of  $Q$ ,  $\partial E / \partial V_i(t)$ . If  $\partial E / \partial V_i(t) < 0$ , then  $V_i(t)$  increases in the next iteration. This

fact is completely reflected in the  $i$ th row of the 2D-BHN since the first neuron in state 0 will change to state 1. Similarly, if  $\partial E / \partial V_i(t) > 0$ , then  $V_i(t)$  decreases, and the last neuron of the  $i$ th row of the 2D-BHN in state 1 will change to state 0.

The pair  $(\mathbf{T}^*, \mathbf{i}^*)$  can be obtained by comparing equations 1.3 and 3.1. They are

$$T_{ij,kl}^* = \begin{cases} T_{ik}\Delta + A, & i = k, \quad j = l - 1, \quad l + 1, \\ T_{ik}\Delta, & \text{otherwise,} \end{cases} \quad (3.4)$$

$$I_{ij}^* = \begin{cases} I_i, & j = 1, \\ I_i - A, & \text{otherwise.} \end{cases} \quad (3.5)$$

Finally, from equations 3.4 and 3.5,  $\hat{\mathbf{T}}$  and  $\hat{\mathbf{i}}$  are

$$\hat{T}_{ij} = \begin{cases} T_{kl}\Delta + A, & k = l, \quad i = j - 1, \quad j + 1, \\ T_{kl}\Delta, & \text{otherwise,} \end{cases} \quad (3.6)$$

$$\hat{i}_i = \begin{cases} I_k, & i = p(k - 1) + 1, \\ I_k - A, & \text{otherwise.} \end{cases} \quad (3.7)$$

$$k = \left\lceil \frac{i}{p} \right\rceil, \quad l = \left\lceil \frac{j}{p} \right\rceil, \quad (3.8)$$

where  $\lceil x \rceil$  is the ceiling function, that is, the function that returns the smallest integer not less than  $x$ . Note that the neurons of the  $i$ th row of  $\hat{Q}^*$  correspond to the neurons  $p(i - 1) + 1$  to  $pi$  of  $\hat{Q}$ .

As a result, the BHN  $\hat{Q}$  defined in equations 3.6 and 3.7 and obtained from the original QHN  $Q$  satisfies theorem 1.

Consequently, and from Bruck and Goodman (1988), if  $\mathbf{v}(t) = \mathbf{v}(t - 1)$ , then it can be concluded that the QHN has reached equilibrium. On the other hand, if  $\mathbf{v}(t) \neq \mathbf{v}(t - 1)$  and  $\mathbf{v}(t) = \mathbf{v}(t - 2)$ , the QHN is oscillating between  $\mathbf{v}(t)$  and  $\mathbf{v}(t - 1)$ .

#### 4 Minimization of the Energy Function

---

Section 3 showed that any QHN operating in a fully parallel mode reaches a stable state or a cycle of length two. Cycles are not as problematic as they may seem, as shown in section 2. The biggest problem is that QHNs do not always reduce the energy from one iteration to the next. Therefore, although QHNs reach a stable state, the energy at that state could be greater than the energy at the initial state, since a stable state is always a local minimum of

the energy but not necessarily the absolute one. This section proves that the energy at any stable state is always less than the energy of the initial state that led to it.

First, the next theorem reveals what happens in the QHN evolution when the energy increases:

**Theorem 2.** *Let  $Q \equiv (T, \mathbf{i})$  be a QHN. If the energy increases from iteration  $t$  to iteration  $t + 1$ , at least one component of the energy gradient changes sign:*

$$\text{If } \Delta E > 0 \rightarrow \exists i, \quad \text{sign} \left( \frac{\partial E}{\partial V_i}(t) \right) = -\text{sign} \left( \frac{\partial E}{\partial V_i}(t + 1) \right), \quad (4.1)$$

$$\Delta E = E(t + 1) - E(t). \quad (4.2)$$

**Proof.** The increment of the energy is

$$\begin{aligned} \Delta E &= \frac{1}{2} \mathbf{v}^T(t) \mathbf{T} \mathbf{v}(t) + \mathbf{v}^T(t) \mathbf{i} - \frac{1}{2} \mathbf{v}^T(t + 1) \mathbf{T} \mathbf{v}(t + 1) - \mathbf{v}^T(t + 1) \mathbf{i} \\ &= \frac{1}{2} \sum_{i=1}^N \Delta_i(t) \left( \frac{\partial E}{\partial V_i}(t) + \frac{\partial E}{\partial V_i}(t + 1) \right). \end{aligned} \quad (4.3)$$

If  $\Delta E > 0$ , then at least one term of the summation of equation 4.3 must be positive. If the  $i$ th term is positive, then

$$\text{If } \Delta_i(t) \geq 0 \rightarrow \frac{\partial E}{\partial V_i}(t) \leq 0 \rightarrow \frac{\partial E}{\partial V_i}(t + 1) \geq 0. \quad (4.4)$$

Thus, from equation 4.4, the energy gradient of neuron  $i$  changes sign.

Some additional conclusions can be drawn from analysis of the proof of theorem 2. From equation 4.3, some neurons can be identified as guilty of the energy increase. As it has been proven, all components of the energy gradient that correspond to the guilty neurons change sign. Thus, the QHN tends to correct the cause of the energy increment since for those neurons,  $\Delta_i(t) + \Delta_i(t + 1) = 0$  (recall equation 1.2). This relevant result hints that the QHN has a good evolution despite sporadic increments of the energy. The following theorem states formally this fact:

**Theorem 3.** *Let  $Q \equiv (T, \mathbf{i})$  be a QHN. Then if  $Q$  reaches a stable state, the energy at this state is less than or equal to the energy of any previous state.*

**Proof.** Let us define  $t_e$  as the iteration where  $Q$  reaches the stable state, that is,  $\Delta_i(t_e) = 0 \forall i$ , and  $t_a < t_e$  as any previous iteration. Similarly to

equation 4.3, the increment of the energy from  $t_a$  to  $t_e$  is

$$\begin{aligned}
 E(t_e) - E(t_a) &= \frac{1}{2} \sum_{t=t_a}^{t_e-1} \sum_{i=1}^N \Delta_i(t) \left( \frac{\partial E}{\partial V_i}(t) + \frac{\partial E}{\partial V_i}(t+1) \right) \\
 &= \frac{1}{2} \sum_{i=1}^N \left[ \Delta_i(t_a) \frac{\partial E}{\partial V_i}(t_a) + \sum_{t=t_a+1}^{t_e-1} (\Delta_i(t-1) + \Delta_i(t)) \frac{\partial E}{\partial V_i}(t) \right. \\
 &\quad \left. + \Delta_i(t_e-1) \frac{\partial E}{\partial V_i}(t_e) \right]. \tag{4.5}
 \end{aligned}$$

Therefore, the energy increment can be divided into  $N$  terms—one per neuron. Each term is further split into products of neuron variations and energy gradients. Now this proof will focus on each one of these products to show that all of them are nonpositive, which proves the validity of the theorem.

The first product,  $\Delta_i(t_a) \partial E / \partial V_i(t_a)$ , is always nonpositive due to equation 1.2. For the last product,  $\Delta_i(t_e-1) \partial E / \partial V_i(t_e)$ , two different cases can be identified. The first case is when the equilibrium is reached provided that  $\partial E / \partial V_i(t_e) = 0$  or  $\Delta_i(t_e-1) = 0$ . Then the last product is, obviously, nonpositive. If  $\partial E / \partial V_i(t_e) \neq 0$ , that means that the  $i$ th neuron is at one of the hypercube extremes, that is,  $V_i(t_e) = 0$  or  $V_i(t_e) = 1$ . Moreover, if  $\Delta_i(t_e-1) \neq 0$ , the  $i$ th neuron has reached the extreme exactly at iteration  $t_e$ . Then  $V_i(t_e-1) = (p-1)\Delta$  and  $\Delta_i(t_e-1) = \Delta$ , or  $V_i(t_e-1) = \Delta$  and  $\Delta_i(t_e-1) = -\Delta$ . In both cases,  $\Delta_i(t_e-1)$  and  $\partial E / \partial V_i(t_e)$  must have opposite signs; if not,  $\Delta_i(t_e) = -\Delta_i(t_e-1)$ , and  $Q$  would not be at the stable state.

For the rest of products, three different cases must be studied. The first case is when  $\Delta_i(t-1)$  and  $\Delta_i(t)$  have the same sign, or  $\Delta_i(t-1) = 0$ ; the second is when  $\Delta_i(t-1)$  and  $\Delta_i(t)$  have different signs—hence, their sum is zero; and the third is when  $\Delta_i(t) = 0$ . Since  $\Delta_i(t)$  and  $\partial E / \partial V_i(t)$  have opposite signs, all the products of the first case are nonpositive. For the second case,  $\Delta_i(t-1) + \Delta_i(t) = 0$ , and thus these products are also nonpositive. Finally, for the last case, if  $\Delta_i(t) = 0$  because  $\partial E / \partial V_i(t) = 0$ , then the products are nonpositive. Second, if  $\partial E / \partial V_i(t) \neq 0$ , then the  $i$ th neuron is at one of the extremes at iteration  $t$ ; and  $\Delta_i(t-1)$  and  $\partial E / \partial V_i(t)$  must have opposite signs; if not,  $\Delta_i(t)$  would not be zero, as demonstrated for the last product. Table 1 summarizes all possible combinations of this second term.

Moreover, from this proof, theorem 3 can be generalized to any iteration  $t$  that satisfies the following condition:

$$\text{If } \sum_{i=1}^N \Delta_i(t-1) \frac{\partial E}{\partial V_i}(t) \leq 0 \rightarrow E(t) \leq E(t_a), \quad \forall t_a < t. \tag{4.6}$$

Table 1: Possible Combinations of the Second Term of Equation 4.5.

Case	Sign of			
	$\Delta_i(t - 1)$	$\Delta_i(t)$	$\partial E / \partial V_i(t)$	$(\Delta_i(t - 1) + \Delta_i(t)) \partial E / \partial V_i(t)$
1	+	+	-	-
	+	0	+	(Impossible combination)
3a	+	0	0	0
3b	+	0	-	$-(V_i(t) = 1)$
2	+	-	+	0
1	0	+	-	-
1	0	0	+	$0 (V_i(t) = 0)$
1	0	0	0	0
1	0	0	-	$0 (V_i(t) = 1)$
1	0	-	+	-
2	-	+	-	0
3b	-	0	+	$-(V_i(t) = 0)$
3a	-	0	0	0
	-	0	-	(Impossible combination)
1	-	-	+	-

## 5 Conclusion

This letter has extended previous studies, analyzing the convergence and stability of QHNs operating in a fully parallel mode. This analysis is interesting because these neural networks are easily implementable and take advantage of the original parallelism of HNNs.

Moreover, this letter has proved that QHNs operating in fully parallel mode always converge to a stable state or to a cycle of length two. Moreover, cycles are not very problematic, as shown in section 2, and this type of network requires many fewer iterations to find a good solution of the MQP compared with CHNs and other QHNs operating in serial mode.

Finally, although the energy does not always decrease from one iteration to the next, the QHN dynamics always tend to decrease the energy, obtaining a stable state with less energy than the initial state of the neural network.

As future work, a deep analysis of cycles would be very interesting. Although the results of section 2 show that cycles do not damage the performance of QHNs for the MQP, they could have severe consequences in other applications. Additionally, a step forward after this letter is the implementation—or an implementation study—of QHNs in digital devices, like FPGAs.

## Appendix

This appendix presents some technical issues about the simulations performed in section 2. The energy function used is Le and Pham

(2005):

$$\begin{aligned}
E = & \frac{A}{2} \sum_{i=1}^N \left( \sum_{j=1}^N V_{ij} - 1 \right)^2 + \frac{A}{2} \sum_{j=1}^N \left( \sum_{i=1}^N V_{ij} - 1 \right)^2 \\
& + \frac{B}{2} \sum_{i=1}^N \sum_{j=1}^N V_{ij} \sum_{1 \leq i-k, j-k \leq N (k \neq 0)} V_{i-k, j-k} \\
& + \frac{B}{2} \sum_{i=1}^N \sum_{j=1}^N V_{ij} \sum_{1 \leq i-k, j+k \leq N (k \neq 0)} V_{i-k, j+k}, \tag{A.1}
\end{aligned}$$

with  $A = B = 1000$ . All elements of the main diagonal of  $\mathbf{T}$  are exactly  $-2A$ , which are not nonnegative (i.e., they are negative). The CHNs were not implemented with the analog circuit but simulated in a computer. The evolution of the analog circuit can be characterized by the following differential equation:

$$\frac{dV_{ij}(t)}{dt} = \begin{cases} 0, & \frac{\partial E}{\partial V_{ij}}(t) > 0, V_{ij}(t) = 0, \\ 0, & \frac{\partial E}{\partial V_{ij}}(t) < 0, V_{ij}(t) = 1, \\ -\alpha_{ij} \frac{\partial E}{\partial V_{ij}}(t), & \text{otherwise,} \end{cases} \tag{A.2}$$

where  $\alpha_{ij}$  are constants that depend on the elements of the analog circuit. This evolution was simulated using Euler's technique:

$$V_{ij}(t+1) = \begin{cases} V_{ij}(t), & \frac{\partial E}{\partial V_{ij}}(t) > 0, V_{ij}(t) = 0, \\ V_{ij}(t), & \frac{\partial E}{\partial V_{ij}}(t) < 0, V_{ij}(t) = 1, \\ V_{ij}(t) - \Delta t \frac{\partial E}{\partial V_{ij}}(t), & \text{otherwise,} \end{cases} \tag{A.3}$$

with  $\Delta t = 10^{-5}$ . This value was selected by a trial-and-error procedure. Lower values make the CHNs require more iterations to converge, whereas greater values make CHNs diverge. Note that the time variable  $t$  is different in equation A.2 than from the rest of the letter, including equation A.3. In equation A.2,  $t$  is continuous since this equation corresponds with the actual evolution of CHNs, which are continuous in time. In the rest of the letter,  $t$  is an integer that represents the current iteration.

## Acknowledgments

---

Part of this work has been performed in the framework of the CELTIC project CP5-013 ICARUS. This work was partially supported by the Spanish Ministry of Industry, Tourism and Trade and the FEDER program

of the European Commission under the project TSI-020400-2008-113 and the Universidad Politécnica de Valencia (PAID-06-08/3301).

## References

---

- Ahn, C. W., & Ramakrishna, R. S. (2004). QoS provisioning dynamic connection-admission control for multimedia wireless networks using a Hopfield neural network. *IEEE Transactions on Vehicular Technology*, *53*, 106–117.
- Bousoño-Calzón, C., & Salcedo-Sanz, S. (2004). A discrete-time quantized-state Hopfield neural network. *Annals of Mathematics and Artificial Intelligence*, *42*, 345–367.
- Bruck, J., & Goodman, J. W. (1988). A generalized convergence theorem for neural networks. *IEEE Transactions on Information Theory*, *34*, 1089–1092.
- Calabuig, D., Monserrat, J., Gómez-Barquero, D., & Lázaro, O. (2006). User bandwidth usage-driven HNN neuron excitation method for maximum resource utilization within packet-switched communication networks. *IEEE Communications Letters*, *10*, 766–768.
- Calabuig, D., Monserrat, J. F., Gómez-Barquero, D., & Cardona, N. (2008). A delay-centric dynamic resource allocation algorithm for wireless communication systems based on HNN. *IEEE Transactions on Vehicular Technology*, *57*, 3653–3665.
- Forti, M., Manetti, S., & Marini, M. (1992). A condition for global convergence of a class of symmetric neural circuits. *IEEE Transactions on Circuits and Systems I*, *39*, 480–483.
- Haykin, S. (1999). *Neural networks: A comprehensive foundation*. Upper Saddle River, NJ: Prentice Hall.
- Hopfield, J. J. (1984). Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of National Academy of Sciences*, *81*, 3088–3092.
- Hopfield, J. J., & Tank, D. W. (1985). “Neural” computation of decisions in optimization problems. *Biological Cybernetics*, *52*, 141–152.
- Joya, G., Atencia, M. A., & Sandoval, F. (2002). Hopfield neural networks for optimization: Study of the different dynamics. *Neurocomputing*, *43*, 219–237.
- Lázaro, O., & Girma, D. (2000). A Hopfield neural-network-based dynamic channel allocation with handoff channel reservation control. *IEEE Transactions on Vehicular Technology*, *49*, 1578–1587.
- Le, T. N., & Pham, C. K. (2005). A new N-parallel updating method of the Hopfield-type neural network for N-queens problem. In *Proceedings IEEE International Joint Conference on Neural Networks* (Vol. 2, pp. 788–791). Piscataway, NJ: IEEE Press.
- Matsuda, S. (1999a). Quantized Hopfield networks for integer programming. *System and Computers in Japan*, *30*, 1–12.
- Matsuda, S. (1999b). Theoretical analysis of quantized Hopfield network for integer programming. *Proceedings IEEE International Joint Conference on Neural Networks* (Vol. 1, pp. 568–571). Piscataway, NJ: IEEE Press.
- Moon, T. K., & Stirling, W. C. (2000). *Mathematical methods and algorithms for signal processing*. Englewood Cliffs, NJ: Prentice Hall.
- Wilson, G. V., & Pawley, G. S. (1988). On the stability of the travelling salesman problem algorithm of Hopfield and Tank. *Biological Cybernetics*, *58*, 63–70.