A Study into the Improvement of Binary Hopfield Networks for Map Coloring

Gloria Galán-Marín[1], Enrique Mérida-Casermeiro[2], Domingo López-Rodríguez[2], and Juan M. Ortiz-de-Lazcano-Lobato[3]

[1] Department of Electronics and Electromechanical Engineering, University of Extremadura, Badajoz, Spain
gloriagm@unex.es

[2] Department of Applied Mathematics, University of Málaga, Málaga, Spain
{merida,dlopez}@ctima.uma.es

[3] Department of Computer Science and Artificial Intelligence, University of Málaga, Málaga, Spain
jmortiz@lcc.uma.es

Abstract. The map-coloring problem is a well known combinatorial optimization problem which frequently appears in mathematics, graph theory and artificial intelligence. This paper presents a study into the performance of some binary Hopfield networks with discrete dynamics for this classic problem. A number of instances have been simulated to demonstrate that only the proposed binary model provides optimal solutions. In addition, for large-scale maps an algorithm is presented to improve the local minima of the network by solving gradually growing submaps of the considered map. Simulation results for several n-region 4-color maps showed that the proposed neural algorithm converged to a correct colouring from at least 90% of initial states without the fine-tuning of parameters required in another Hopfield models.

1 Introduction

The k-coloring problem is a classic NP-complete optimization problem. The four color theorem states that any map drawn on a plane or sphere can be colored with four colors so that no two areas which share a border have the same color. The proof of this conjecture took more than one hundred years [1]. In 1976, Appel and Haken provided a computer-aided proof of the four-color theorem [2].

A powerful neural network for solving the map-coloring problem was presented by Takefuji and Lee [3]. The capability of the neural algorithm was demonstrated by solving examples of Appel and Haken's experiments through a large number of simulation runs. Remarkable solutions for many other combinatorial optimization problems have been presented by applying Takefuji and Lee's model, showing that it performs better than the best known algorithms [4,5,6]. Takefuji and Lee found discrete neurons computationally more efficient than continuous neurons [3]. Hence, they usually apply the continuous dynamics of the analog

B. Beliczynski et al. (Eds.): ICANNGA 2007, Part II, LNCS 4432, pp. 98–106, 2007.

Hopfield model with binary neurons. However, it has been recently demonstrated that this model does not always guarantee the descent of the energy function and can lead to inaccurate results and oscillatory behaviors in the convergence process [7,8,9,10].

In contrast, recently we have presented two binary Hopfield networks with discrete dynamics that always guarantee and maximize the decrease of the energy function. In the first one [11], a new input-output function is introduced into the binary Hopfield model with an asynchronous activation dynamics. Simulation results show that this sequential network converges to global optimal solutions for the n-queens problem [11]. However, since the operation of this model is based on the notion of single update, the required number of iteration steps for convergence is increased in proportion to the size of the problem. It led us to design a new binary neural network, the optimal competitive Hopfield model (OCHOM), based on the notion of group update [7]. It has been observed that the computation time is decreased even 100 times for large-scale networks comparing to the sequential model presented in [11]. In addition, performance comparison trough massive simulation runs showed that for some problems the OCHOM is much superior to Takefuji and Lee's model in terms of both the solution quality and the computation time [7,8].

Recently, Wang and Tang [12] have improved the OCHOM by incorporating stochastic hill-climbing dynamics into the network. Simulation runs show that for some problems this algorithm obtains better solutions than the OCHOM, though the computation time is increased.

In this paper we study the performance of the binary neural networks through the four-color map problem. Despite the remarkable solutions obtained for some combinatorial optimization problems, simulation results show that the OCHOM network does not provide a global minimum solution for the map-coloring problem. Note that Joya et al. [13] proved that the Hopfield model with discrete dynamics never reached a correct solution for the k-colorability problem, while continuous dynamics succeeded to obtain a correct colouring. However, as pointed in [13], one mayor problem with the continuous model is that there is no analytical method to obtain the parameters of the network.

The mayor advantage of the OCHOM is that the search space is greatly reduced without a burden on the parameter tuning. However, it becomes a disadvantage for the map-coloring problem, where reducing the magnitude of the search space can easily bring on the problem of the local minimum convergence. It leads us to apply the sequential binary Hopfield model where more states of the network are allowed since every neuron can be activated on every step.

We have applied the binary sequential Hopfield model with the discrete input-output functions proposed by other authors [15,16]. It is confirmed by computer simulations that these networks never succeed to obtain a correct colouring. However, applying our discrete function [11] the sequential Hopfield network is capable of generating exact solutions for the four-color problem. On the other hand, simulation runs for large-scale maps show that the percentage of random initial states that do not converge to a global minimum is considerably increased.

Even for Hopfield networks with continuous dynamics it has been reported that with extremely large maps it is necessary to dynamically adjust the parameters so as to avoid local minima [14], where no method is given for this task.

We have modified the binary sequential network applying a hill-climbing term. However, we have found that this technique do not guarantee global minimum convergence for large-scale maps if the initial state is not a "correct" one. Hence, we have developed a method to avoid this difficulty without the fine-tuning of parameters required in another Hopfield models. The proposed algorithm solves a growing submap with the sequential network and uses as the initial state this optimal solution instead of a random binary state. Simulation runs in up to the 430-country map taken from the example of Appel and Haken's experiment illustrate the effectivity and practicality of this neural approach.

2 Network Architecture

Let H be a binary neural network (1/0) with N neurons, where each neuron is connected to all the other neurons. The state of neuron i is denoted by v_i and its bias by θ_i, for $i = 1, \ldots, N$; ω_{ij} is a real number that represents the interconnection strength between neurons i and j, for $i, j = 1, \ldots, N$. Note that we do not assume that self-connections $\omega_{ii} = 0$, as in the traditional discrete Hopfield model. Considering discrete-time dynamics, the Liapunov function of the neural network is given by:

$$E(k) = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij} v_i(k) v_j(k) + \sum_{i=1}^{N} \theta_i v_i(k) \tag{1}$$

where k denotes discrete time. The inputs of the neurons are computed by the Hopfield's updating rule: $u_i(k) = \sum_{j=1}^{N} \omega_{ij} v_j(k) - \theta_i$. We assume now that the network is updated asynchronously, that is, only one neuron i is selected for updating at time k. Hence, we have a sequential model where $\Delta v_i \neq 0$, $\Delta v_j = 0$, $j = 1 \ldots n$, $j \neq i$, and the energy change is:

$$\Delta E(k) = E(k+1) - E(k) = -\Delta v_i(k)[u_i(k) + \frac{\omega_{ii}}{2}\Delta v_i(k)] \tag{2}$$

Since we have binary outputs $v_i \in \{0, 1\}$, it follows from (2) that:

- For $\Delta v_i = 1$, $\Delta E(k) \leq 0$ if and only if $u_i(k) \geq -\frac{\omega_{ii}}{2}$
- For $\Delta v_i = -1$, $\Delta E(k) \leq 0$ if and only if $u_i(k) \leq \frac{\omega_{ii}}{2}$

From these conditions we get that the energy is guaranteed to decrease if and only if the input-output function is:

$$v_i(k+1) = \begin{cases} 1 \; \textit{if} \; v_i(k) = 0 \; \textit{and} \; u_i(k) \geq -\frac{\omega_{ii}}{2} \\ 0 \; \textit{if} \; v_i(k) = 0 \; \textit{and} \; u_i(k) < -\frac{\omega_{ii}}{2} \\ 0 \; \textit{if} \; v_i(k) = 1 \; \textit{and} \; u_i(k) \leq \frac{\omega_{ii}}{2} \\ 1 \; \textit{if} \; v_i(k) = 1 \; \textit{and} \; u_i(k) > \frac{\omega_{ii}}{2} \end{cases} \tag{3}$$

For $\boldsymbol{\omega_{ii}} \geq \mathbf{0}$ the above expression reduces to:

$$v_i(k+1) = \begin{cases} 1 & if\ u_i(k) > |\frac{\omega_{ii}}{2}| \\ 0 & if\ u_i(k) < -|\frac{\omega_{ii}}{2}| \\ change & if\ -|\frac{\omega_{ii}}{2}| \leq u_i(k) \leq |\frac{\omega_{ii}}{2}| \end{cases}$$

and for $\boldsymbol{\omega_{ii}} < \mathbf{0}$ becomes to:

$$v_i(k+1) = \begin{cases} 1 & if\ u_i(k) \geq |\frac{\omega_{ii}}{2}| \\ 0 & if\ u_i(k) \leq -|\frac{\omega_{ii}}{2}| \\ no\ change & if\ -|\frac{\omega_{ii}}{2}| < u_i(k) < |\frac{\omega_{ii}}{2}| \end{cases} \quad (4)$$

Sun [15] proposed a generalized updating rule (GUR) for the binary Hopfield model updated in any sequence of updating modes. In the case of sequential mode this generalized updating rule is equivalent to the function proposed by Peng et al. in [16] and very similar to (4). Since these functions are only valid when $\omega_{ii} < 0$, it shows that (3) is not an instance of them.

Observe that the function presented in [15,16] only differs from (4) in the case that $u_i(k) = |\frac{\omega_{ii}}{2}|$ and $v_i(k) = 0$ and in the case of $u_i(k) = -|\frac{\omega_{ii}}{2}|$ and $v_i(k) = 1$. In these cases we have $\Delta E = 0$ if we change the state of the neuron. Simulation runs in the four-coloring problem, a problem with $\omega_{ii} < 0$, show that only if we allow the network to evolve to another states with the same energy it is possible to reach the global minimum. For this reason our function (4) enables the network to generate a correct colouring. However, applying the function proposed in [15,16] the network is always trapped in unacceptable local minima.

3 The Proposed Algorithm for the Map-Coloring Problem

The neural network is composed of $N \times K$ binary neurons, where N is the number of areas to be colored and K is the number of colors available for use in coloring the map. The binary output of the *ikth* neuron $v_{ik} = 1$ means that color k is assigned to area i, and $v_{ik} = 0$ otherwise. Hence, the energy function is defined:

$$E = B \sum_{i=1}^{N} (\sum_{k=1}^{K} v_{ik} - 1)^2 + \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j \neq i}}^{N} \sum_{k=1}^{K} a_{ij} v_{ik} v_{jk} \quad (5)$$

where $B > 0$ is a constant which specifies the relative weighting of the first term and $A = [a_{ij}]$ is the adjacency matrix which gives the boundary information between areas, that is, $a_{ij} = 1$ if areas i and j have a common boundary, and $a_{ij} = 0$ otherwise (see fig. 1). The first term in the energy function (5) becomes zero if one and only one neuron in each row has 1 as the output, and so a unique color is assigned to each area of the map. The second term vanishes when all neighboring areas do not have the same color. By comparing the energy function

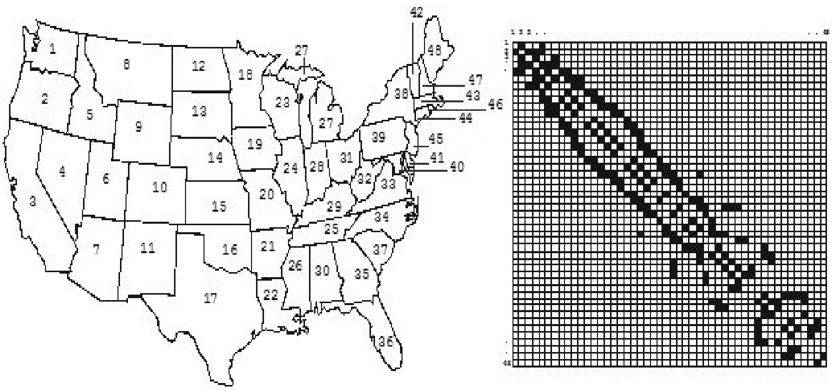

Fig. 1. The U.S. continental map and its adjacency matrix given by a 48 × 48 array. Black and white squares represent 1 and 0 values, respectively.

defined (5) and the Hopfield energy function (1), the connections weights and the biases are derived. If we substitute them in the Hopfield's updating rule then:

$$u_{ik} = -\theta_{ik} + \sum_{j=1}^{N}\sum_{s=1}^{K} \omega_{ik,js} v_{js} = 2B - 2B\sum_{s=1}^{K} v_{is} - 2\sum_{\substack{j=1 \\ j \neq i}}^{N} a_{ij} v_{jk} \tag{6}$$

Observe that the neural self-connection is $\omega_{ik,ik} = -2B$ for all the neurons and then, according to (2), the energy change on every step is:

$$\Delta E = -u_{ik}\Delta v_{ik} + B(\Delta v_{ik})^2 \tag{7}$$

Simulation runs show that for medium-sized maps the proposed binary sequential model obtains a correct colouring from randomly generated initial states. However, for large-scale maps it is usually necessary a mechanism for improving local minima. The proposed neural algorithm is based upon the observed fact that the network can find more easily an exact solution for a given map using as the initial state an optimal solution of a submap of it, rather than using as the initial state a random binary matrix V.

In our algorithm on step n the neural network colors a map with n regions, using $n \times K$ binary neurons and considering the adjacency matrix $n \times n$. On step $n+1$, one region is added to the map and the network solves the $(n+1)$-map using $(n+1) \times K$ binary neurons and considering the correct adjacency matrix $(n+1) \times (n+1)$. The algorithm introduces the optimal solution for the n-map problem obtained on step n as the initial state for the $(n+1)-$map problem by completing with a $1 \times K$ random binary vector the row $n+1$. This does not mean that the optimal solution that we obtain on step $n+1$ must always include the solution obtained on step n, since the network can evolve to a different state.

Note also that the size of the network is modified with time and automatically adjusted to the size of the considered map.

Observe that in the algorithm NI_{max} denotes the maximum number of iteration steps for the time out procedure for solving the n-map. It means that if the network is trapped in a local minimum for the map with n regions, the algorithm automatically forces the network to solve the map with $n+1$ regions, and so on. For the 4-color problem, the network with $n \times 4$ neurons is extended to a network with $(n+1) \times 4$ neurons, and so there is an increment of the energy function. Then, since the energy and the number of neurons are increased, there are more possible states with less or equal energy for the network to evolve in order to escape from local minima. The following procedure describes the algorithm proposed for solving the N-map K-coloring problem, in which we start with $n = 1$ or $n = \alpha \leq N$, where α is the number of regions in a solved submap:

1. Initialize the values of all the neuron outputs v_{ik}, for $i = 1$ to n and $k = 1$ to K, by randomly choosing 0 or 1.
2. Solve the n-map K-coloring problem:
   (a) Extract from A $(N \times N)$ the adjacency matrix $A_n(n \times n)$ for the n-map problem and set the number of iteration steps $NI = 0$.
   (b) Evaluate the initial value of the energy function (5).
   (c) Select randomly a neuron ik.
   (d) Compute u_{ik}, the input of neuron ik, by eq. (6).
   (e) Update the neuron output v_{ik} by the input-output function (4).
   (f) Compute the energy change by eq. (7) and the new value of E.
   (g) Increment the number of iteration steps by $NI = NI + 1$.
   (h) Repeat from step 2.c until $E = 0$ or $NI = NI_{max}$.
3. If $n = N$ then terminate this procedure, else go to step 4.
4. Add a $(1 \times K)$ random binary row vector to the optimal matrix $V = [v_{ik}]_{n \times K}$ obtained in 2.
5. Increment n by $n = n + 1$ and go to 2.

Observe that, if the network is still trapped in a local minimum when we reach $n = N$, that is, the full number of regions of the map, we can add imaginary regions to the real map until the network reaches the global minimum.

4 Simulation Results

We have tested the different binary Hopfield networks on the n-region 4-color maps solved by Takefuji and Lee in [3], that is, the U.S. continental map which consists of 48 states (see fig. 1) and the maps taken from the experiments of Appel and Haken [2]. Computational experiments were performed on an Origin 2000 Computer (Silicon Graphics Inc.) with 4 GBytes RAM by Matlab. For every map and network, we carried out 100 simulation runs from different randomly generated initial states. Simulation results showed that the OCHOM network [7,8] never reached a correct solution. Also, the binary model proposed in [15,16] was always trapped in unacceptable local minima for all the considered maps.

Initially, we have applied the proposed sequential model without the algorithm that solves gradually growing submaps. To prevent the network to assign no color to areas with a large number of neighbors, we strengthen the first term of the energy function by taking the constant value of the coefficient $B = 2$. Simulation runs for the U.S. map showed that the network converged to exact solutions, as the ones represented in fig. 2, from 95% of the random initial states. Then, we have found a few initial states from which the network is trapped in local minima. We have modified the network applying to our model a hill-climbing term. However, simulation runs in all the considered maps show that this technique does not either guarantee global minimum convergence.

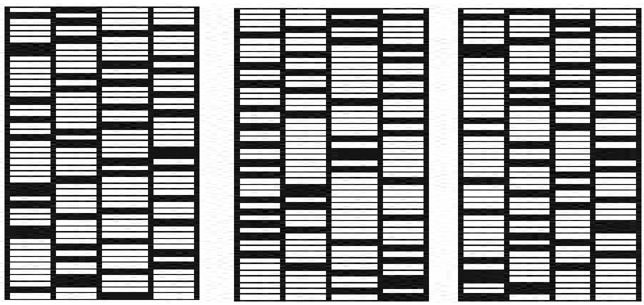

Fig. 2. Three different solutions $[V]_{48\times4}$ of the 48-state U.S. map 4-color problem provided by the sequential network. Black and white rectangles indicate 1 and 0 outputs, respectively, where the output $v_{ik} = 1$ means that color k is assigned to state i.

We consider now a 210-country map taken from Appel and Haken's experiments and extract arbitrary submaps of 50, 70, 100 and 150 countries, where each one includes the one before. For the 50-map the network converged to an exact solution from 94 % of the random initial states, and for the 70-map from 92% of the random initial states. When we consider the maps with $100, 150$ and 210 countries it is confirmed that for these similarly difficult maps the percentage of random initial states that converges to a correct colouring decreases with the size of the map. Therefore, for the 430-country map taken from Appel and Haken's experiments it is a rather difficult task finding randomly an initial state from which the network reaches a global minimum. Hence, it becomes necessary for extremely large maps to complete the network with the proposed algorithm to find adequate initial states from submaps.

When we applied the complete algorithm described in Sect. 3 with $n = 1$, the sequential model converged to an exact solution for the U.S. map from 100% of the initial states for a total of 100 runs. Also, for the 210-map of Appel and Haken the percentage of network simulations that produced a correct coloring was 100%. Observe that the network gradually colors all the extracted submaps, in this case 209 maps. Finally, when we apply this algorithm to the 430−map of Appel and Haken, 90 % of the network simulations provided a correct coloring for a total of 100 runs. This percentage can be increased if we add imaginary regions to the real map as described in Sect. 3.

5 Conclusions

A study into the performance of different binary Hopfield networks through the four-coloring problem has been presented. Simulation results show that both the optimal competitive Hopfield model [7,8] and the sequential Hopfield model presented in [15,16] never reached a correct colouring. However, the proposed binary sequential Hopfield model is capable of generating exact solutions for medium-sized maps. Nevertheless, simulation results also show that the percentage of random initial states that converges to a correct colouring decreases with the size of the map when we consider similarly difficult maps. Hence, a neural algorithm is presented for large-scale maps to help the sequential model to find adequate initial states by solving submaps of the considered map. A number of instances have been simulated showing that this network provides an efficient and practical approach to solve the four-coloring problem even for large-scale maps without a burden on the parameter tuning.

References

1. Saaty,T., Hainen, P.: The four color theorem: Assault and Conquest. Mc Graw-Hill (1977).
2. Appel, K., Haken, W.: The solution of the four-color-map problem, Scientific American, Oct. (1977) 108-121.
3. Takefuji, Y., Lee, K. C.: Artificial neural networks for four-colouring map problems and K-colorability problems. IEEE Trans. Circuits Syst. **38** (1991) 326-333.
4. Funabiki, N., Takenaka, Y., Nishikawa, S.: A maximum neural network approach for N-queens problem. Biol. Cybern. **76** (1997) 251-255.
5. Funabiki, N., Takefuji, Y., Lee, K. C.: A Neural Network Model for Finding a Near-Maximum Clique. J. of Parallel and Distributed Computing **14** (1992) 340-344.
6. Lee, K. C., Funabiki, N., Takefuji, Y.: A Parallel Improvement Algorithm for the Bipartite Subgraph Problem. IEEE Trans. Neural Networks **3** (1992) 139-145.
7. Galán-Marín, G., Muñoz-Pérez, J.: Design and Analysis of Maximum Hopfield Networks. IEEE Transactions on Neural Networks **12** (2001) 329-339.
8. Galán-Marín, G., Mérida-Casermeiro, E., Muñoz-Pérez, J.: Modelling competitive Hopfield networks for the maximum clique problem. Computers & Operations Research **30** (2003) 603-624.
9. Wang, L.: Discrete-time convergence theory and updating rules for neural networks with energy functions. IEEE Trans. Neural Networks **8** (1997) pp. 445-447.
10. Tateishi, M., Tamura, S.: Comments on 'Artificial neural networks for four-colouring map problems and K-colorability problems'. IEEE Trans. Circuits Syst. I: Fundamental Theory Applicat. **41** (1994) 248-249.
11. Galán-Marín, G., Muñoz-Pérez, J.: A new input-output function for binary Hopfield Neural Networks. Lecture Notes in Computer Science, Vol. 1606 (1999) 311-20.
12. Wang, J., Tang, Z.: An improved optimal competitive Hopfield network for bipartite subgraph problems. Neurocomputing **61** (2004) 413-419.

13. Joya, G., Atencia, M. A., Sandoval, F.: Hopfield neural networks for optimization: study of the different dynamics. Neurocomputing **43** (2002) 219-237.
14. Dahl, E. D.: Neural Network algorithm for an NP-Complete problem: Map and graph coloring. Proc. First Int. Joint Conf. on Neural Networks **III** (1987) 113-120.
15. Sun, Y.: A generalized updating rule for modified Hopfield neural network for quadratic optimization. Neurocomputing **19** (1998) 133-143.
16. Peng, M., Gupta, N. K., Armitage, A. F.: An investigation into the improvement of local minima of the Hopfield network. Neural Networks **9** (1996) 1241-1253.