

A generalization of the Hopfield model for the graph isomorphism problem

GLORIA GALÁN-MARÍN¹, DOMINGO LÓPEZ-RODRÍGUEZ², ENRIQUE MÉRIDA-CASERMEIRO²

¹ Department of Mechanical, Energetic and Materials Engineering
University of Extremadura

Badajoz

² Department of Applied Mathematics

University of Málaga

Málaga

SPAIN

gloriagm@unex.es, dlopez@ctima.uma.es, merida@ctima.uma.es

Abstract: - Isomorphism identification between graphs is an important NP-complete problem with many science and engineering applications. Although excellent progresses have been made towards special graphs, no known polynomial-time algorithm for graph isomorphism has been found for general graphs. In this paper a generalization of the Hopfield neural network for isomorphism identification between general graphs is proposed. Simulation results show that this model is much superior to recently presented neural networks for this problem. The effectiveness of the resultant network does not seem to be decreased as the size of the graph is increased. This allows us to solve graph isomorphism problems with a big number of vertices, while many recently presented approaches only present results for graphs with up to 15 vertices.

Key-Words: - Graph isomorphism, Neural networks, Kinematic chains, Optimization, Computational dynamics, Multivalued models.

1 Introduction

A lot of time and effort has been devoted to developing a reliable and computationally efficient technique for the detection of isomorphism among graphs. Applications of the graph isomorphism problem [1] include: cyclic sequential circuit, organic molecule structures, mechanical structures, etc. As recently pointed out [2], most published algorithms still leave a lot to be desired in different aspects, such as simplifying procedure of identification and adapting automatic computation. Recently proposed eigenvector approaches have attracted a lot of attention [2,3,4], since they are methods developed for automatic computation. However, all these recently published algorithms only present results for graphs with up to 15 vertices and do not report description of computation time.

Neural networks are an alternative direction for graph isomorphism detection, since they allow parallel computation which may be needed when we have a big number of vertices in a graph. An example of this is the molecule DNA sequence in biotechnology. Besides, this kind of iteration-based approaches may provide a means to compare two graphs on an approximate basis which is useful in DNA sequence where hundreds of vertices are in a graph. Kong et al

[5] presented a model for graph isomorphism detection based on the continuous Hopfield model [6]. Recently, we proposed a binary discrete neural network for this problem [7], showing that it is much superior to the previously presented network of Kong et al [5] in terms of the computation time and the interpretation of solutions. However, the effectiveness of the binary neural algorithm [7] decreases as the number of links is increased. In fact, we show in this paper that this binary neural approach requires a prohibitively long computation time to solve a graph isomorphic problem with 28 vertices. For this reason, in this work we apply a generalization of the Hopfield model [8], instead of the binary model, designing new computational dynamics to solve the graph isomorphism problem. Simulation results show that a test graph problem with 28 vertices is solved in seconds on a conventional PC.

2 The Proposed Network for the Graph Isomorphism Problem

In this section, a generalization of the Hopfield network presented in [8] is applied to solve the graph isomorphism problem. The proposed model is

characterized by the neuron outputs taking value in a discrete set, denoted by $M=\{m_1, m_2, \dots, m_L\}$. If v_i is the state of neuron i , then $v_i \in M$. Associated to each state vector V , an energy function, similar to Hopfield's, can be defined:

$$E(V) = -\frac{1}{2} \sum_i \sum_j w_{i,j} f(v_i, v_j) + \sum_i \theta_i(v_i) \quad (1)$$

where $W=(w_{i,j})$ is the synaptic weight matrix (expressing the connection strength between neurons), $f:M \times M \rightarrow R$ is the so-called similarity function (since $f(v_i, v_j)$ measures the similarity between the outputs of neurons i and j), and $\theta_i:M \rightarrow R$ is the generalization of the biases $\theta_i \in \bullet$, present in Hopfield's model. The aim of the network is to achieve a stable state corresponding to a global or local minimum of the energy function (1). This network is a generalization of the Hopfield's neural network since it includes this model as a particular case. It suffices to define $M=\{0,1\}$ (unipolar case) or $M=\{-1,1\}$ (bipolar neurons), $f(x,y)=xy$ and $\theta_i(x)=\theta_i x$, being θ_i the threshold or bias used in Hopfield's model.

Note that two graphs are said to be isomorphic if there exists a one to one equivalent relation between their vertices and edges that preserve the incidence. As demonstrated in [4], the graphs represented by the adjacency matrices A and B are isomorphic, if and only if $B = PAP^T$, where P is an orthogonal permutation matrix such that $P^T=P^{-1}$. This operation matrix P has the feature that on each column and row there is only one element for 1 and all the others for 0. Then, a solution of the graph isomorphism problem can be represented by a neuron state V with n elements that is a permutation of the rows and columns of the adjacency matrix $A(n \times n)$. Therefore, we consider $M=\{1,2,\dots,n\}$, that is, only state vectors V representing permutations of the numbers $\{1,2,\dots,n\}$ are the feasible states. This means that V is an abbreviate notation for the permutation matrix $P=(p_{i,j})$ such that $p_{i,j}=1$ if, and only if, $v_i=j$, otherwise it is 0. With this notation, the permuted matrix will be $A'=(a_{v_i, v_j})$. Given two graphs with adjacency matrices A and B , we define a new energy function to be minimized that can be expressed as a function that measures the differences between the permuted matrix A' and B :

$$\begin{aligned} E(V) &= \frac{1}{2} \sum_i \sum_j (a_{v_i, v_j} - b_{i,j})^2 = \frac{1}{2} \sum_i \sum_j (a_{v_i, v_j}^2 - 2a_{v_i, v_j} b_{i,j} + b_{i,j}^2) \\ &= \frac{1}{2} \sum_i \sum_j (a_{v_i, v_j} - 2a_{v_i, v_j} b_{i,j} + b_{i,j}) = -\sum_i \sum_j (a_{v_i, v_j} b_{i,j}) + \text{const} \end{aligned}$$

This energy function achieves its global minimum, $E=0$, when $A'=B$, that is, A and B represent

isomorphic graphs. If we skip the constant term, and define the synaptic weights $w_{i,j}=2b_{i,j}$ and the similarity function $f(x,y)=a_{x,y}$, we obtain the identification between the energy function (1) and the objective function of the problem. With this identification, computational dynamics are proposed to solve the problem. Updating schemes must verify that network states are always feasible solutions, that is, permutations of $\{1,2,\dots,n\}$. Thus, we have considered that if neurons p and q are marked for updating, then $v_p'=v_p(t+1)=v_q(t)$, $v_q'=v_q(t+1)=v_p(t)$, and $v_i'=v_i(t+1)=v_i(t)$ for $i \neq p, q$. The input (synaptic potential) of neuron p associated to this update is $u = -(\Delta E)_{p,q} = -\sum_i (\Delta_{i,p} + \Delta_{i,q})$, where $\Delta_{i,j}=w_{i,j}(f(v_i, v_j)-$

$f(v_i', v_j')$). Particularly, $\Delta_{i,p}=2b_{i,p}(a_{v_i, v_p} - a_{v_i, v_q})$ for $i \neq p, q$, and analogously for $\Delta_{i,q}$. Note that $\Delta_{p,q}=0$.

Observe that a Hopfield-type network can only ensure that, if we reach the energy value $E=0$, then the two graphs are isomorphic. Then, a criterion must be applied to conclude that two graphs are non-isomorphic. As in [5,7] we consider that if the network does not reach the value $E=0$ after N_{max} iterations, this corresponds to two graphs that are not isomorphic.

3 Simulation Results

In this section we present results obtained by the proposed model. For comparison, we also present results given by the binary Hopfield model [7]. Both algorithms were implemented on a 3 GHz Pentium IV PC with 512 MBytes RAM by MATLAB. As case studies, we show in figure 1 some examples proposed in the recent bibliography to test methods developed for automatic computation in the graph isomorphism problem [3,9]. Table 1 presents the average computation time required by each algorithm to solve the test problems, where the results were obtained for a total of 1000 independent runs. Note that for the non-isomorphic test problem 1(b), the algorithm performs $N_{max} = 100$ iterations since it is detected that the network does not reach $E=0$. Numerical results show that the proposed generalization of the Hopfield neural network is much superior to the binary model [7] in terms of the computation time for all the test problems. Observe that the proposed model takes an average time of 0.07 seconds to solve the selected example with 28 vertices, while the binary model requires a prohibitively long computation time.

Table 1. Comparison of the average

computation time needed for solving the test problems using the recently presented binary Hopfield model (BH) [7] and the proposed generalization of the Hopfield model (GH).

	BH	GH
Fig. 1 (a) (isomorphic with 14 vertices)	87.1 s	0.03 s
Fig. 1 (b) (non-ison graphs with 15 vertices)	217.41 s	1.17 s
Fig. 1 (c) (isomorphic with 28 vertices)	> 24 (hours)	0.07 s

4 Conclusion

In this paper we design a new generalization of the Hopfield neural network that allows a simplified representation of the graph isomorphism problem. Simulation results for the considered case studies show that the computation time is shorter than that of the recently presented neural network for this problem [7]. Besides, the effectiveness of the proposed network does not seem to be decreased as the size of the graph is increased. This allows us to solve graphs with a big number of vertices, where other approaches only present results for graphs in up to 15 vertices [2-5,7]. The chief merit of this method is its computational simplicity and efficiency. Thus, the approach is specially suitable for automatic computation since it rapidly provides solutions without a burden on the parameter tuning.

References:

[1] R.J. Wilson, L.W. Beineke (Eds.), 1979, "Applications of Graph Theory," Academic Press, New York.

[2] Cubillo, J.P., Wan, J.B., 2005, "Comments on mechanism kinematic chain isomorphism identification using adjacent matrices," *Mech. Mach. Theory* 40 (2), pp. 131-139.

[3] He, P.R., Zhang, W.J., Li, Q., 2005, "Some further development on the eigensystem approach for graph. isomorphism detection," *J. Franklin Inst. – Eng. Appl. Math.* 342, pp. 657–673.

[4] Sunkari, R.P., Schmidt L.C., 2006, "Reliability and efficiency of the existing spectral methods for isomorphism detection," *ASME J. Mech. Des.* 128 (6), pp. 1246-1252.

[5] Kong, F.G., Li, Q., Zhang W.J., 1999, "An artificial neural network approach to mechanism kinematic chain isomorphism identification," *Mech. Mach. Theory* 34 (2), pp. 271-283.

[6] Hopfield, J. J. and Tank, D. W., 1985, "Neural computation of decisions in optimization problems," *Biol. Cybern.* 52, pp. 141-152.

[7] Galan-Marin, G., Merida-Casermeiro, E., Lopez-Rodriguez, D., 2007, "Improving Neural Networks for Mechanism Kinematic Chain Isomorphism," *Neural Proc. Letters* 26, pp. 133-143.

[8] Merida-Casermeiro, E., Muñoz Perez, J., 2002, "Mrem: An associative autonomous recurrent network," *J. Intelligent and Fuzzy Systems* 12, pp. 163 – 173.

[9] He, P.R., Zhang, W.J., Li, Q., 2001, "A quadratic form-based approach to identification of kinematic chains in structural analysis and synthesis of mechanisms", *Proc. 21st ASME Computers and Information in Engineering Conference, Pittsburgh, Pennsylvania, DETC2001/DAC-21067.*

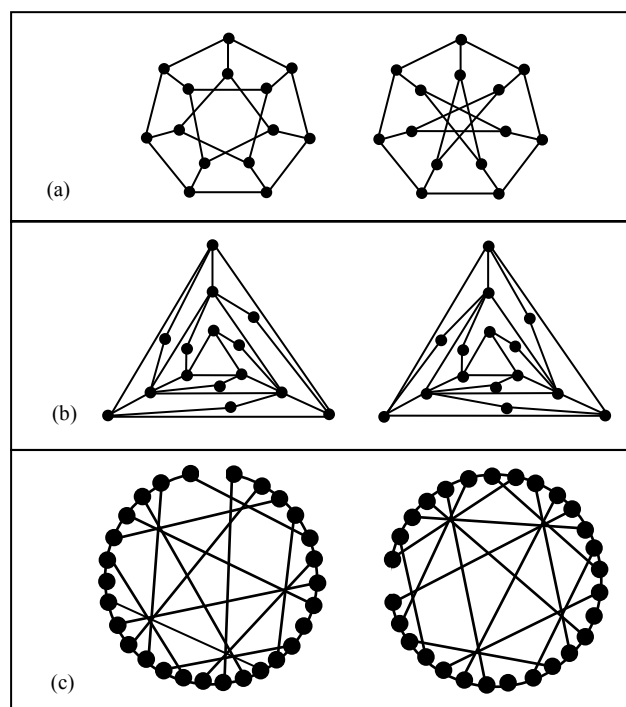


Figure 1. Different pairs of isomorphic and non-isomorphic graphs.