A Noisy Chaotic Neural Network for Solving Attributed Relational Graph Matching Problem

Kushan Ahmadian [1], Abbas Golestani [1], M.R.Jahed Motlagh [1] and Nasser Mozayani [1]
[11]Computer faculty, Iran University of Science and Technology, Tehran, Tehran, Iran

Abstract - *In this paper we propose a new gradual noisy chaotic neural network (MP-NCNN) to solve the NP-complete attributed relational graph matching problem. These graphs are very important in pattern matching applications and the noisy chaotic behavior of the proposed method which avoids getting trapped in local minima, yields in better results and hence it is more effective approach in comparison with previous methods. The performance of the proposed method has been evaluated through several attributed relational graphs with different permuted vertices and also with different vertex numbers. The obtained results show that the proposed method outperforms previous approaches including HNN, CNN and TCNN methods.*

Keywords: ARG, chaotic neural networks, Optimization, NCNN.

1 Introduction

Robustness and flexibility of Attributed graphs (ARGs) make them powerful data structures for object representation in pattern recognition [1]. These graphs are used in many pattern matching applications and by acquiring these models, the problem of the problem seeking optimal, partial and inexact homomorphism of a model in a scene turns into matching ARGs of scene and model. This problem is generally referred as the attributed relational graph matching problem and it's proved to be NP-hard which means there is no known deterministic algorithm to solve this problem in polynomial time.

Recently many new algorithms have been proposed to solve ARGMP and although HNN [2] proved to an effective algorithm to solve ARGMPs it usually being trapped by local minima. To overcome the shortcomings of classic Hopfield neural networks, chaotic dynamics has been applied to HNN networks by many authors.

In [3] Aihara et al, has introduced a chaotic neural network which exhibits chaotic behavior using a negative self-coupling and gradually removing it. The model is based on the chaotic behavior of some biological neurons and has been successfully applied to several optimization problems e.g. TSP [4][5] and their method outperformed the Hopfield neural network in both the efficiency and solution quality and comparing to other neural networks , their approach significantly increased the probability of finding near-optimal solutions. Afterward Nozawa [6], introduced a new CNN by applying the Euler discretization in order to simulate continues and their results from the TSP method proofed the efficiency of their algorithm. Moreover, Chen and Aihara [7], introduced a chaotic simulated annealing approach which was firstly, based on a transiently chaotic phase and second, a convergence phase which tended to find the global minimums. Their method has been successfully applied to the TSP and machine maintenance problem [7]. Chen and Aihara also showed the existence of strange attractors and network stability conditions [8] suggested the dynamical phenomenon of crisis-induced intermittency to be the underlying mechanism for the chaotic switching among the minima.

Wang and smith [9], proposed another discretzed neural network Hopfield network, Continues-Time Continues-Output (CTCO-HNN), and by varying the timestamp which is a control parameter to control the network dynamics and it is similar to temperature parameter of classic stochastic simulated annealing. By gradually decreasing of timestamp, the network dynamics changes to a reverse bifurcation process which provides a transiently chaotic behavior. The N-Queen case study in [10] proofs the ability of this method to solve optimization problems.

Zhou et al. [11] added a chaotic noise using a chaotic time series generated by the Henon map, to the network as an external approach, in contrast to the internal approaches described previously in which the chaotic behavior is generated internally. They applied their method to the 100-city TSP problem and the obtained results showed superior optimization ability compared to the Boltzmann machine.

2 Attributed Relational Graphs (ARGs)

If we consider a^t as the type and a^v as the value, then an attribute is an ordered couple of (a^t, a^v), and an attribute set is a set of all couples which belong to s particular feature primitive. In addition a vertex attribute set can be defined as set of attributes associated with an ARG and each attribute is a set of $\{...,(a_n^{t,x},a_n^{v,x})\}$. Moreover an edge also has an attribute set associated with it which is called the edge attribute set. If we assume V as the vertex attribute set and E as the set of edge attribute set, then an ARG of and object can be denoted as G={V,E}. Usually $G_s=\{V^s,E^s\}$ is used to denote a scene and $G_m=\{V^m,E^m\}$ to denote a model. ARG matching is to find V'^s a subset of V^s, and V'^m, a subset of V^m, such that there is a mapping for every scene vertex in V'^s in the model vertex V'^m, and whenever a pair of distinct scene and model vertices has the one-to-one mapping, the corresponding edges should also be mapped.

In [12], Suganthan, has proposed an HNN-based algorithm for the ARG matching problem but it has been proofed that their HNN-based algorithm will halt when it reaches a local optimal result. This method has been defined as follow:

$$\left\{\begin{array}{c} v_{xi}=g(u_{xi}) \\ \dfrac{du_{xi}}{dt}=\sum\limits_{y,j}C_{xi,yi}-A_c\sum\limits_{j\neq i}v_{xj} \end{array}\right\} \quad (1)$$

$$E=-\sum_{x,y,i,j}C_{xi,yj}v_{xi}v_{yi}+A_c\sum_{x,i,j\neq i}v_{xj}v_{xi} \quad (2)$$

Here A_c is the relative strength of the soft constraint and $C_{xi,yj}$ is the compatibility measure. Vertex v_x in the scene and vertex v_i in the model are said to be matched if v_{xi} is larger than the threshold.

3 ARG matching using NCNN

This algorithm is effective in ARG matching but when it is trapped in a local optimum, the performance degrades. The proof for this issue can be found in [13]. Since this kind of network can easily be trapped in local minima, stochastic simulated annealing [14] has been combined with the HNN.

In [14], Wang and Tian, proposed a new simulated annealing method by injecting a stochastic noise into the TCNN. They showed that their algorithm is more likely to find optimal solutions in comparison with CSA. Their method has been successfully applied to TSP and channel assignment problem and especially in large scale TSPs the NCNN achieved much better performance. The NCNN model is introduced as follows:

$$x_{jk}(t)=\frac{1}{1+e^{-y_{jk}(\tau)/\varepsilon}} \quad (3)$$

$$y_j(t+1)=ky_{jk}(t)-z(t)\lfloor x_{jk}(t)-I_0\rfloor+n(t)+\left\{\sum_{i=1,i\neq j}^{N}\sum_{l=1,l\neq k}^{M}w_{jkil}x_{jk}(t)+I_{jk}\right\} \quad (4)$$

$$z(t+1)=(1-\beta_1)z(t) \quad (5)$$

$$n(t+1)=(1-\beta_2)n(t) \quad (6)$$

Where

x_{jk} : Output of the single neuron jk,

y_{jk} : Input of neuron jk

w_{jkil} : Connection weight between neurons jk and il where $w_{jkil}=w_{iljk}$, $w_{ikjk}=0$

$$\left\{\sum_{i=1,i\neq j}^{N}\sum_{l=1,l\neq k}^{M}w_{jkil}x_{jk}(t)+I_{jk}\right\}=-\frac{\partial E}{\partial x_{jk}} \quad (7)$$

I_{jk} : Input bias of neuron jk

k: Damping factor of nerve membrane $0\leq k\leq 1$

α : Positive scaling parameter for inputs

β_1 Damping factor for neural self-coupling $0\leq\beta_1\leq 1$

β_2 : damping factor for stochastic noise $0\leq\beta_2\leq 1$

z(t): Self-feedback connection weight, $z(t)\geq 0$

I_0 : Positive parameter

ε Steepness parameter of the output function

E: Energy function

n(t): Random noise of neurons, with a uniform distribution

A[n]: amplitude of noise n

From the above equations, the TCNN model for a single neuron can be obtained as follows:

$$x(t) = \frac{1}{1 + y^{-\frac{y(\tau)}{\varepsilon}}} \qquad (8)$$

$$y(t+1) = ky(t) + \alpha I_i - z(t)[x(t) - I_0] + n(t) \qquad (9)$$

$$z(t+1) = (1 - \beta_1) z(t) \qquad (10)$$

$$A[n(t+1)] = (1 - \beta_2) A[n(t)] \qquad (11)$$

By combining (8) and (9) the, equations can be rewritten as:

$$y(t+1) = ky(t) + \alpha I_i - z(t)\left[\frac{1}{1+e^{-y(t)/\varepsilon}} - I_0\right] + n(t) \qquad (12)$$

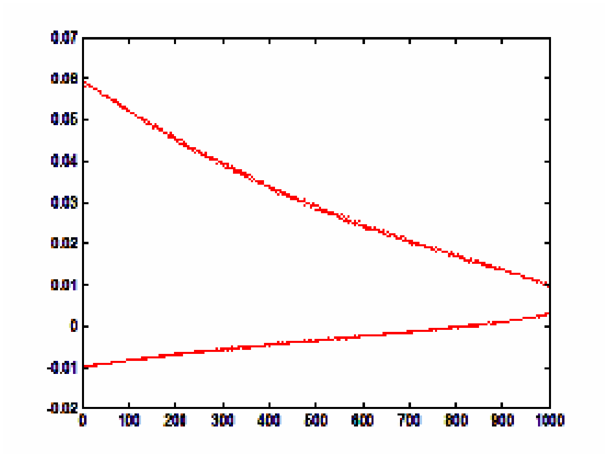
Different values for the constant parameters of this equation make different neurodynamics. In order to achieve best results we have chosen the following parameters for the dynamics of neurons.

$$\alpha = 0.015, \beta_1 = 0.001, \varepsilon = 0.004, I_0 = 0.65, \beta_2 = 0.0001$$

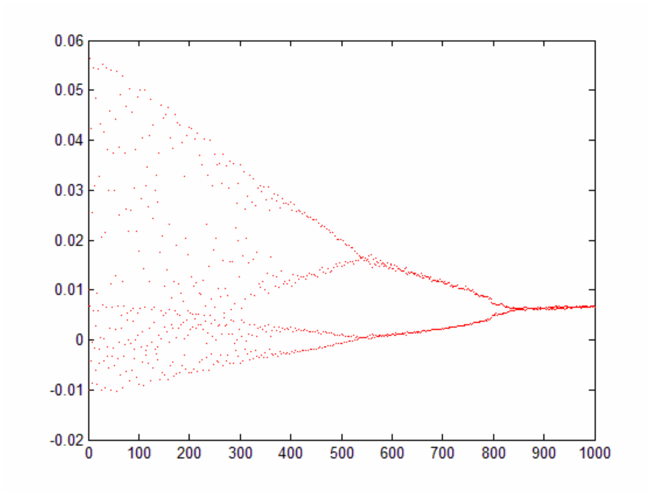
Since Performance of ARG-matching is influenced by the local optimal results, The MP-NCNN model which has been proofed to be more effective in comparison with the classic TCNN method. The dynamics of the proposed method for ARG-matching is as follows:

$$y_{jk}(t+1) = y_{jk}(t) - z(t)\left\lfloor x_{jk}(t) - I_0 \right\rfloor + n(t) + \alpha\left[\sum C_{jk} x_{il}(t) + A_c \sum_{j \neq i} v_{xi}(t)\right] \qquad (13)$$

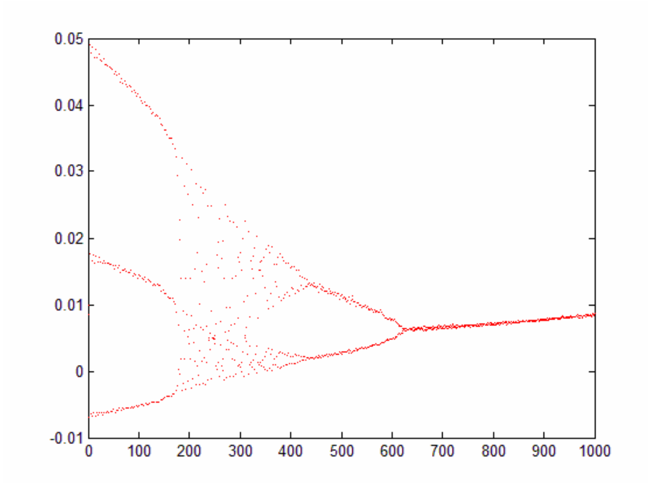
Different behaviors of Figure 1 shows the dynamics of a single neuron for different values of k in which the x axis is the time stamp t, and y axis is the output of the neuron x(t). As it can be seen in these figures, the larger values of k, results in better bifurcations. In our model to achieve a better chaotic behavior we have chosen k to be 0.8.



(a)



(b)



(c)

Figure 1. Dynamics of a single neuron for different k parameters. $\alpha = 0.015$, $\beta_1 = 0.001$, $\varepsilon = 0.004$, $I_0 = 0.65$, $\beta_2 = 0.0001$. (a) k=0.2, (b) k=0.5 and (c) k=0.8

4 Results

In order to evaluate performance of the proposed method we chose to test it over 50-vertex, 100-vertex and 150-vertex ARGs with different values of connectivity ranging from 50% to 100%. In order to obtain isomorphic graphs, we chose 100 ARGs with random permuted vertices. In this model success was in matching each of each vertex of the model and the scene. To compare the results we compared the proposed method's results with classic HNN and TCNN. The obtained results can be seen in table 1 in which the first column is the measure of connectivity for the ARGs, the second, third and forth columns are the rate of success for HNN, TCNN and NCNN for each number of vertices respectively.

Table 1. Experimental results from comparison of the proposed method with HNN and TCNN

Measure of connectivity	No. of vertices	HNN	TCNN	NCNN
1.0	50-vertex	92.4%	96.7%	**97.1%**
	100-vertex	90.7%	95.3%	**97.0%**
	150-vertex	87.5%	91.1%	**93.5%**
0.9	50-vertex	91.9%	**96.4%**	96.0%
	100-vertex	91.0%	94.9%	**95.3%**
	150-vertex	89.3%	92.0%	**93.9%**
0.8	50-vertex	87.9%	**95.9%**	95.4%
	100-vertex	85.2%	90.5%	**93.7%**
	150-vertex	85.1%	89.1%	**93.2%**
0.7	50-vertex	86.0%	91.1%	**92.4%**
	100-vertex	85.1%	88.5%	**90.9%**
	150-vertex	83.5%	86.4%	**88.0%**
0.6	50-vertex	81.6%	88.3%	**91.4%**
	100-vertex	80.3%	**88.0%**	87.9%
	150-vertex	78.9%	85.6%	**87.4%**
0.5	50-vertex	80.4%	87.3%	**90.4%**
	100-vertex	77.9%	85.0%	**88.5%**
	150-vertex	77.3%	84.9%	**88.2%**

From the results it can be seen that performance of the proposed algorithm is considerably better in most cases (except 3 of them, 0.9 connectivity and 50 vertices, 0.8 connectivity and 50 vertices and 0.6 connectivity with 100 vertices) compared to HNN and TCNN. This is due the ability of the proposed method to escape the local minima.

5 Conclusions

In this paper we presented a noisy chaotic neural network for solving the ARG mapping problem. We evaluated the algorithm through several attributed relational graphs with different vertex numbers. We also compared our results with HNN and TCNN. Unlike HNN which usually becomes trapped in local minima, the algorithm of this paper tends to find the global solution in most cases. Compared to TCNN it also showed a better performance. The obtained results show that in most cases the proposed method finds the best solution with minimal energy. The results prove that the proposed algorithm is an efficient method for solving ARGMP

6 References

[1] M.A Eshera, K.S. Fu, “An image understand system using attributed system using attributed symbolic representation and inexact graph matching”, IEEE Transactions on PAMI, Vol.8, No.5, pp.604-617, 1986.

[2] J.J. Hopfield, D.W. Tank. “Neural computation of decisions in optimization problems”, Biological Cybernetics, Vol 52, pp.141-152, 1985.

[3] K. Aihara, T. Takabe, and M. Toyoda, “Chaotic neural networks,” *Phys. Lett. A*, vol. 144, nos. 6, 7, pp. 333–340, 1990.

[4] T. Yamada, K. Aihara, and M. Kotani, “Chaotic neural networks and the travelling salesman problem,” in *Proc. Int. Joint Conf. Neural Networks*, pp. 1549–1552, 1993.

[5] M. Hasegawa, T. Ikeguchi, T. Matozaki, and K. Aihara, "Solving combinatorial optimization problems by nonlinear neural dynamics," in *Proc. IEEE Int. Conf. Neural Networks*, vol. 6, pp. 3140–3145, 1995.

[6] H. Nozawa, "A neural-network model as a globally coupled map and applications based on chaos," *Chaos*, vol. 2, no. 3, pp. 377–386, 1992.

[7] L. Chen and K. Aihara, "Chaotic simulated annealing by a neural network model with transient chaos," *Neural Networks*, vol. 8, no. 6, pp. 915–930, 1995.

[8] L. Chen and K. Aihara, "Chaos and asymptotical stability in discrete time neural networks," *Phys. D*, vol. 104, pp. 286–325, 1997.

[9] L. Wang and K. Smith, "On chaotic simulated annealing," IEEE Trans. Neural Net., vol. 9, pp. 716–718, 1998.

[10] T. Kwok, K. Smith, and L. Wang, "Incorporating chaos into the Hopfield neural network for combinatorial optimization," World Multiconf. Systems, Cybernetic. Informatics, vol. 1, pp. 659–665, 1998.

[11] D. Zhou, K. Yasuda, and R. Yokoyama, "A method to combine chaos and neural-network based on the fixed point theory," *Trans. Inst. Elect. Eng. Japan*, vol. 117-C, no. 5, pp. 599–608, 1997.

[12] P.N. Suganthan, E.K. Toeh, and D.P. Mital, "Pattern Recognition by graph matching using Potts MFT theory network", Pattern Recogniton, Vol. 28, No. 7, pp.997-1009, 2001.

[13] S. Gu, S. Yu, "A chaotic neural network for the attributed relational graph matching problem in pattern recognition", Intl. Symposium of intelligent multimedida, , pp. 695-698, 2004.

[14] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983.

[15] L.Wang and F. Tian, "Noisy chaotic neural networks for solving combinatorial optimization problems," in *Proc. Int. Joint Conf. Neural Networks (IJCNN 2000)*, Como, Italy, Jul. 24–27, vol. 4, pp. 37–40, 2000.