Proceedings of the 9th International Conference on Neural Information Processing (ICONIP'02), Vol. 3
Lipo Wang, Jagath C. Rajapakse, Kunihiko Fukushima, Soo-Young Lee, and Xin Yao (Editors)

A SIMPLE LEARNING ALGORITHM FOR GROWING RING SOM AND ITS APPLICATION TO TSP

Hiroki Sasamura, Ryuji Ohta and Toshimichi Saito

Dept. of EECE, Hosei University, Tokyo, 184-8584 Japan. *tsaito@k.hosei.ac.jp*

ABSTRACT

This paper presents a simple learning algorithm for self-organizing maps having ring topology and considers its application to TSP. The algorithm has only two control parameters and the map structure can grow by occasional inspection of the learning history. Some experimental results suggest that this simple algorithm enables the map to extract feature of input data and to find almost optimal solution of the TSP.

1. INTRODUCTION

This paper presents a simple learning algorithm for self-organizing maps having ring topology (ab. RSOMs). Our approach differs from former work in this direction as no ring structure with a fixed number of a cell. The structure of the RSOM is updated based on the winning number of each cell and states of its direct neighbors. The structure of the RSOM can grow by occasional inspection of learning history of each cell. The algorithm has only two control parameters: learning rate and the inspection interval. We then apply the algorithm to a traveling salesperson problem (ab. TSP). Some experimental results suggest that this simple algorithm enables RSOM to extract feature of input data and to find almost optimal solution of the TSP provided the control parameters are selected suitably. Such almost optimal solutions are hard to be found without the growing structures.

Self-organizing maps (ab. SOMs [1]) are known as typical unsupervised learning systems which can extract features from input data. The SOMs have variety of applications including speech recognition and data compression. In order to develop function of SOMs, it is important to control the structure and size adaptively and some interesting learning algorithms have been presented [2]-[6]. In the algorithms, basically, the cell structures grow or reduce depending on the learning history. The growing and reducing are effective to improve the SOMs' functions provided a number of control parameters are adjusted suitably. Such algorithms have been applied to data visualization [2], vector quantization [2] [4], knowledge discovery [6] and so on. Also, SOMs having one-dimensional topology have been applied to traveling salesperson problems (ab. TSPs) [7] [8] and skeletonization [9]. However, neither of these "revised" algorithms has superiority over the original Kohonen's algorithm from the viewpoint of simplicity. Our algorithm is much simpler than the existing "revised" algorithms and can realize growing SOMs with interesting functions. This paper may give a first step to construct simple algorithms for flexible SOMs with growing and/or reducing functions.

2. ALGORITHM

Basic structure of the RSOM is shown in Fig.1 where X is a 2-dimensional input space and A is a discrete space of cells. The RSOM has ring topology consisting of N cells: each cell is connected to the both sides cells. Let $N_i \equiv \{i-1, i, i+1\}$ be the neighbors set of the cell i where i is modulus N. The cell i is characterized by a n-dimensional synaptic vector w_i and a signal counter C_i for inspection of the learning history. The complete algorithm for our model which we call "Growing Ring SOM" is given by the following :

Step 1: Initialization
Let t be a discrete learning time starting from 0 and let $N(t)$ be the number of the cells at time t. At $t = 0$, we give initial conditions $N(0)$, $C_i(0)$ and $w_i(0)$, where $i \in \{1, \cdots, N(0)\}$. For an example in Fig. 2, $N(0) = 3$, $C_i(0) = 0$ and $w_i(0)$ corresponds to either apex of the triangle.

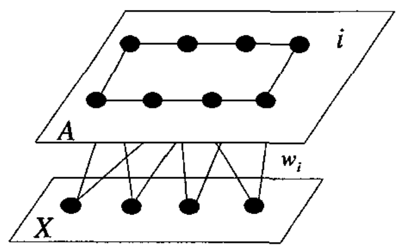

Figure 1: SOM having ring topology

Step 2: Input signal
Apply input signal $x \in X$ according to some distribution.

Step 3: Determination of winner cell
Cell c is selected as a winner at time t if its synaptic vector $w_c(t)$ is the closest to the input x at time t.

$$\|x(t) - w_c(t)\| = \min_i \|x(t) - w_i(t)\| \qquad (1)$$

where $\|\cdot\|$ denotes the Euclidean vector norm.

Step 4: Update of synaptic vectors and counters
According to Equation (2), synaptic vectors of the winner and its neighbors are updated and the other synaptic vectors are preserved at time t.

$$w_i(t+1) = \begin{cases} w_i(t) + \alpha(x(t) - w_i(t)) & \text{if } i \in N_c \\ w_i(t) & \text{otherwise} \end{cases} \qquad (2)$$

where i is modulus $N(t)$. The learning rate α is the first control parameter of this algorithm. The signal counter of the winner is updated and the other signal counters preserve their values at time t.

$$C_i(t+1) = \begin{cases} C_i(t) + 1 & \text{if } i = c \\ C_i(t) & \text{otherwise} \end{cases} \qquad (3)$$

Step 5: Insertion of a cell
Our algorithm inserts one novel cell at every T_{int} learning times hereby the map can grow. At $t = nT_{int}$, n is a positive integer, we determine one cell q whose signal counter has the maximum value.

$$C_q(nT_{int}) \geq C_i(nT_{int}) \text{ for all } i.$$

If there exist plural maximum counter values we randomly select one of them. This is an inspection of the learning history. Then we take either neighbor of f with larger distance in an input space. This is a cell f given by

$$f = \begin{cases} q-1 & \text{if } \|w_{q-1} - w_q\| \geq \|w_{q+1} - w_q\| \\ q+1 & \text{otherwise,} \end{cases}$$

where f is modulus $N(t)$. A novel cell r is inserted between q and f. The synaptic vector of r is initialized as

$$w_r = 0.5(w_q + w_f) \qquad (4)$$

Counter values of q and r are re-assigned as

$$C_q(t+1) = 0.5C_q(t), \; C_r(t+1) = 0.5C_q(t). \qquad (5)$$

The insertion interval T_{int} is the second control parameter of this algorithm. After the insertion, the number of cells increases: let $N(t+1) = N(t) + 1$.

Step 6: Termination
Let $t = t + 1$. If $t < T_{max}$ then go to step 2, otherwise the learning is terminated, where T_{max} is a learning limit.

3. SIMULATION RESULTS

To consider the ability of this algorithm, we apply the algorithm to the input space for uniform distribution on a circle ($n = 2$, $T_{max} = 20000$). The "Growing Ring SOM" can grow without distortion. This algorithm has only two control parameters and is much simpler than existing algorithms for growing and reducing SOMs . However, this algorithm can extract features of input data as suggested by Fig. 2 and by numerical experiments in the next section. Without the growing structure, the Ring SOM must make distortions as shown in Fig.3.

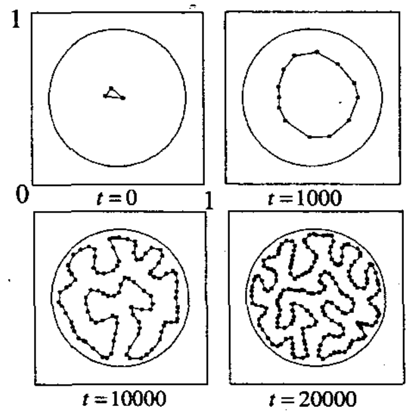

Figure 2: Growing structure for uniform distribution on a circle ($\alpha = 0.1, T_{int} = 100, N(0) = 3$)

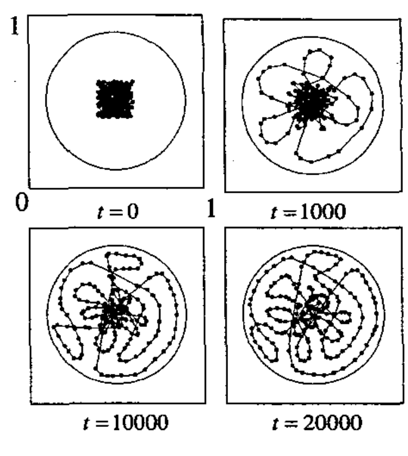

Figure 3: SOM without growing structure for uniform distribution on a circle ($\alpha = 0.1, T_{int} = \infty, N(0) = 200$)

4. APPLICATION TO TSP

We apply the algorithm to a TSP with 48 cities at http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/
At $t = 0$, we give initial conditions:

$$N(0) = 3, \;\; C_i(0) = 0,$$

and $w_i(0)$ are apexes of the triangle in Fig. 4 (left top). The two control parameters are selected as

$$\alpha = 0.1, \;\; T_{int} = 100.$$

Location of each city corresponds to an input and is applied randomly to the RSOM. As learning time goes the RSOM grows and forms a tour route. At $t = 20000$, the RSOM has 200 cells and the learning is terminated. Then we search the closest cell from each city. Every city can obtain the distinct closest cell hereby the tour route can be determined because the RSOM has ring topology (see Fig. 5). The result in Fig. 4 is just 1.4 percent longer than the optimum route shown in the web site: we can find almost optimal route by this simple algorithm. We can not find such almost optimal tour route without the growing structure. In fact, as we use RSOM with fixed cells number ($N(0) = 200$), the cells structure is twisted and the result is chaotic as shown in Fig. 6. Fig. 7 and Fig. 8 show the dependence of the insertion interval T_{int} and learning rate α on the tour length. In the experiments, the final cells number is 200 and 100 trials are executed for each value of T_{int} and α where initial conditions and input data differ for each trial. The redundancy is given for averaged and the shortest tour length for each T_{int} and α. Referring to the averaged graph (bold one), $T_{int} = 100$ and $\alpha = 0.1$ is a suitable value for this TSP. It might be developed into automatic parameters adjustment.

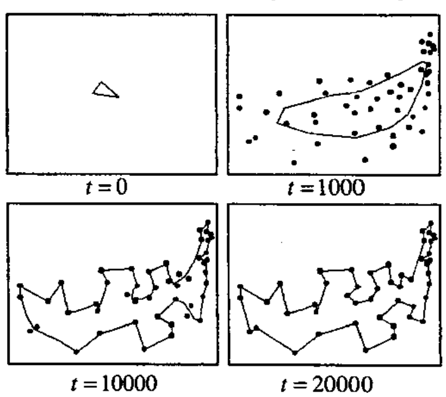

Figure 4: Application to TSP with 48 cities. ($\alpha = 0.1$, $T_{int} = 100$)

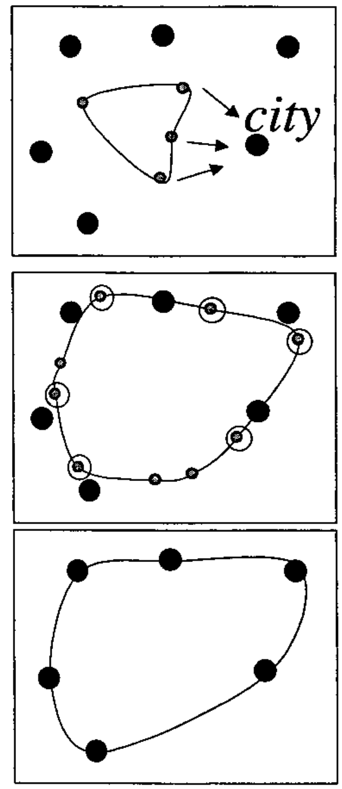

Figure 5: Algorithm for Growing Ring SOM-TSP

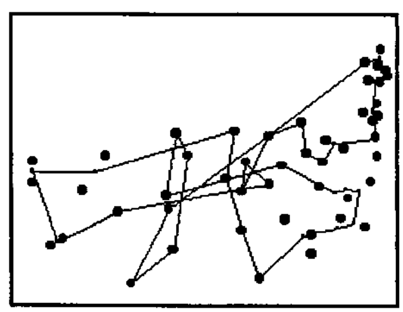

Figure 6: Cells structure at $t = 20000$ by algorithm without growing ($N(0) = 200$, $\alpha = 0.1$, $T_{int} = \infty$). Input data are the same as Fig. 4.

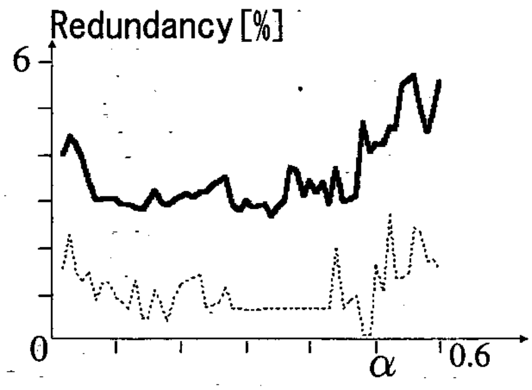

Figure 7: Redundancy for averaged (bold) and the minimum (dot) tour lengths for TSP with 48 cities in Fig.4 (T_{int} = 100, #cells= 200, $\alpha_{max} = 0.6$).

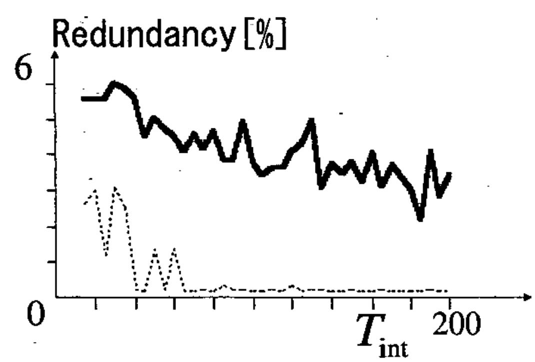

Figure 8: Redundancy for averaged (bold) and the minimum (dot) tour lengths for TSP with 48 cities in Fig.4 ($\alpha = 0.1$, #cells= 200, $T_{max} = 200$).

5. CONCLUSIONS

We have presented a simple learning algorithm for growing RSOM. The algorithm has only two control parameters and is applicable to TSP. Basic experimental results suggest efficiency of the algorithm. Future problems include automatic adjustment of the learning parameters, more detailed evaluation of TSP solutions and analysis of the learning process.

6. REFERENCES

[1] T. Kohonen, "Self-Organized formation of topologically correct feature maps," Biological Cybernetics, 43, pp. 59-69 (1982)

[2] B. Fritzke, "Growing cell structures - a self-organizing network for unsupervised and supervised learning," Neural Networks, 7, pp.1441-1460 (1994)

[3] B. Fritzke, "Growing neural gas network learns topologies" Advances in Neural Information Processing Systems, 7, pp. 625-632 (1995)

[4] S.Kawahara and T.Saito, An adaptive self-organizing algorithm with virtual connection, Proc. of ICONIP, pp.338-341 (1997)

[5] R. Ohta and T. Saito, "A growing self-organizing algorithm for dynamic clustering," Proc. of IJCNN, pp.469-473 (2001)

[6] D. Alahakoon, S. K. Halganmuge and B. Srinivasan, Dynamic self-organizing maps with controlled growth for knowledge discovery, IEEE Trans. Neural Networks, 11, 3, pp. 601-614 (2000)

[7] B. Angeniol, G. de La C. Vaubois and J. Y. Le Texierr, "Self-organizing feature maps and the traveling salesman problem" Neural Networks, 1, pp. 289-293 (1988)

[8] K. Fujimura, H. Tokutaka and M. Ishikawa, "Performance of improved SOM-TSP algorithm for traveling salesman problem of many cities," Trans. IEE Japan, 119-C, 7, pp.875-882 (in Japanese, 1999)

[9] R. Singh, V. Cherkassky and N. Papanikolopoulos, "Self-organizing maps for the skeletonization of sparse shapes," IEEE Trans. Neural Networks, 11, 1, pp. 241-248 (2000)