A Recurrent Neural Network having Auxiliary and Constraint Neurons Applied to Traveling Salesman Problem

Yoshihiro TOMIKAWA
Research & Development Dept.
Yoshida Kogyo K.K. (YKK)
200, Yoshida, Kurobe, 938 Japan

Kenji NAKAYAMA
Dept. of Electrical and Computer Eng.
Faculty of Tech., Kanazawa Univ.
2-40-20, Kodatsuno, Kanazawa, 920 Japan

Abstract:

Traveling Salesman Problem(TSP) is to determine the shortest route for going around several cities. This problem is a kind of NP-complete problems and difficult to solve. In this paper, the method to solve the TSP problem using a recurrent neural network is proposed. This method employs the network model which use an auxiliary neuron and some constraint neurons in addition to the original recurrent neural network in which the connection weights represent the TSP problem. This model can escape the local minimum state of the network by passing through the path on the new space created by the auxiliary neuron.

This model was applied to the 5 cities TSP problems. As the results, it was found that it can detect the shortest routes for those problems. This model is a time continuous model and its behavior is of analogue. Therefore, the simulation took a lot of CPU time. However, by realizing such an analogue device, the solution time will be improved.

1. INTRODUCTION

Neural networks are recently applied to many types of combinatorial optimization problems. These problems are NP-complete and can't be solved by linear programming. Traveling Salesman Problem(TSP) is known to be one of combinatorial optimization problems and difficult to solve. However, since the solution of this problem can be applied to many other applications, pattern matching, optimization control and so on, many methods to solve this problem have been designed [1][2][3][4][5][6].

Hopfield and Tank[3] indicated that a recurrent neural network can find the plausible solution by replacing the TSP problem to the minimization problem of the energy function of the neural network. Their network changes the state of the network discretely so as to decrease the network energy and the stable state of the network gives the minimum solution. On the other hand, Rumelhart et al [4] suggested that the minimum energy of a recurrent network can be found by the continuous state changes. However, it is known that these methods can't detect the global minimum of the energy function because of local minimum.

Then the approaches such as Boltzmann machine[5] and Chaos neural network[6], which permit the temporary increase of the network energy by adding noise, was applied to this problem. But the results of these approaches were stochastic and were not guaranteed to be the real global minimum.

In this paper, the new model is proposed in order to escape the local minimum and to detect the global minimum state of the recurrent neural network. This model uses an auxiliary neuron and increase the degree of freedom of the network instead of increasing the energy.

2. ACTION OF MODEL

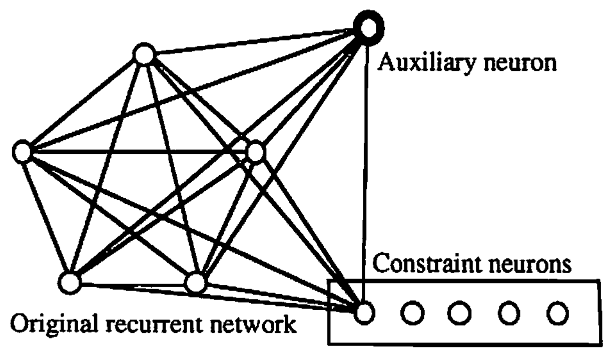

Fig.1 Network Structure Of Proposed Model

Figure 1 shows the structure of the proposed model. It consists of the original recurrent neural network, some constraint neurons and an auxiliary neuron. The original recurrent neural network is shown as the pentagon including 5 neurons but the neurons may increase according to an application.

In order to account for the behavior of the network, let us consider the energy function of the proposed network except the constraint neurons. This energy function is given by the following equation,

$$E(x,u)=1/2x^tWx+th^tx+x^tqu+1/2au^2+bu, \qquad (1)$$

W: Weight matrix of the original network,
th : Threshold vector of the original network,
q: Connection weight vector from the auxiliary neuron to the original network,
a: Recursive connection weight of the auxiliary neuron,
b: Threshold of the auxiliary neuron.

In this equation, the first two terms indicate the energy function of the original network. Therefore, if the state could be found on the space of the proposed network where u=0, it would be equal to the state of the original network. The vector q and the scalers a and b aren't related to the original network. Therefore, they can be controlled freely.

The state change of the proposed network is assumed to be time continuous. If the state changes of the state vector x and the activation of the auxiliary neuron u is given by the following equations,

$$\tau \frac{dx}{dt} = -(Wx + th + qu) \tag{2}$$

$$\tau \frac{du}{dt} = -(au + b + q^t x) \qquad (\tau > 0),$$

the time derivative of Eq.(1) always become negative and the energy $E(x,u)$ always decrease. This state change is one of methods of steepest descent.

Figure 2 shows the energy distribution of the proposed model. In this figure, the u=0 plane indicates the energy distribution of the original network. The surface painted by black color shows the energy distribution of Eq.(1). Our purpose is to detect the minimum state on the u=0 plane, that is to find the point B in Fig.2. If state changes would be restricted only on the u=0 plane, the state B might not be detected only by the decreasing energy method because there is no monotonous descent energy path from the point A to the point Band the state can not change from A to B.

However, by increasing degree of freedom of the state using an auxiliary neuron, it becomes possible to change the state from A to B. If there would be the point out of the u=0 plane which has the medium energy level between A and B and the energy function would be monotonous along the paths from it to A and B, the network could change the state from A to B by passing through that point. Fortunately, the energy function of Eq.(1) is a quadratic form and it has a single saddle point. A quadratic form is monotonous along the straight path from an arbitrary point to the saddle point and satisfied with the previous condition.

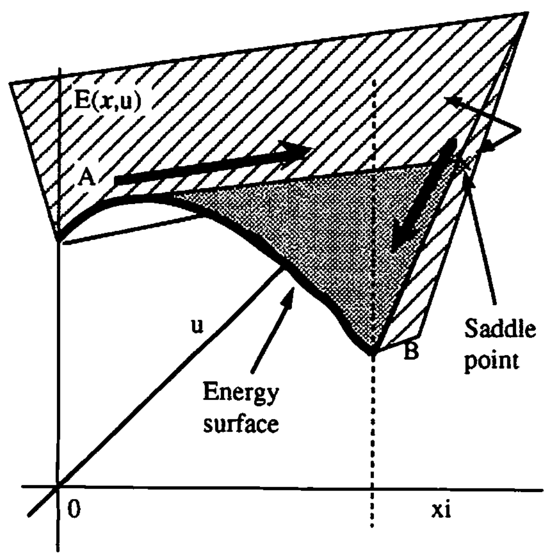

Fig.2 Energy Distribution Of Proposed Model And Minimum Search

Besides, constraint neurons are employed to restrict the network behavior. The auxiliary neuron should be used to help the state change of the network but should not become active. Then the state of the network should be controlled by the constraint neurons. They are assumed to act the wall and to enclose the movable region, as the constraint surfaces in Fig.2.

In Fig.2 only 2 allowable states are shown. However, for the real application, network would have many neurons and many allowable states. In order to detect the minimum energy state among these many states, the energy level of the saddle point should be set on slightly higher value than that of the least energy state, which is unknown. However, by lowering the energy level of the saddle point gradually until the auxiliary neuron become active, the minimum energy state could be detected just before the auxiliary neuron become active. In the simulation of this paper, in order to remove the unnecessary calculations, the different method is employed. However, the purpose is same.

3. PROCESSING IN MODEL

The proposed model uses continuous state change because state must pass through the saddle point. Therefore, the differential method is used for the simulation of the network behavior. In this section, the processing for the simulation are explained.

3.1 Determination of Saddle point.

In order to induce the network to detect the global minimum of the energy function, the vector q and the scalers, a and b must be determined so that the saddle point is on the u=1 plane. Since all the partial differentials of the energy function are 0 at the saddle point, if the saddle point is at $(x_0, 1)$ and the energy level is E_0, the scalers, a and b and the vector q can be determined as follows.

$$q = -(Wx_0 + th), \tag{3}$$

$$a = x_0^t W x_0 + 2th^t x_0 - 2E_0,$$

$$b = 2E_0 - th^t x_0.$$

It is important that the connection weight vector q is irrelevant to the energy level E_0. This means that the energy level of the saddle point can be controlled by processing only the auxiliary neuron.

3.2 State Change (Update)

The state change of the proposed network is given by Eq.(2). This equation is for the continuous state change. For computer simulation, the time derivatives of Eq.(2) are replaced to the time increments, Δx and Δu for small time and the increments are synchronously added to the state vector x and the activation value u.

This state change is assumed to control the state on the energy surface. If the state is on the constraint surface, the constraint neurons contribute to the state of the network. These effects are described later.

3.3 Action of Constraint neurons

Constraint neurons are introduced to the proposed model in order to avoid the undesirable state of the network. They are defined that when they become active, they push back the state of network so that they become inactive. The total influence to the state of the network by constraint neurons is given by the following equation,

$$\frac{dy}{dt} = -\sum_k \boldsymbol{R}_k c_k \tag{4}$$

$$c_k = 1(\boldsymbol{R}_k^{\,t}\, y + r_k)$$

$\boldsymbol{R}_k$: Connection weight vector of the k-th constraint neuron,

r_k : Threshold of the k-th neuron,

$1(\,)$: step function.

In this equation, $y = \{y_0, y_1, \text{--}\}$ is the state vector of the network including the original network and the auxiliary neuron. c_k indicates the activation of the constraint neuron. Step function $1(\,)$ classifies the states to active region and inactive region. $\boldsymbol{R}_k^{\,t}\, y + r_k = 0$ indicates the boundary between both regions. The constraint vector $\boldsymbol{R}_k = \{R_{k0}, R_{k1}, \text{---}\}$ is the normal vector of the boundary. Thus, if the state would be changed along the direction of the vector $\boldsymbol{R}_k$, the state could go back to the inactive region with the least distance.

For simulation, since it takes CPU time to change the state according to the time derivative of Eq.(4), the method which directly moves the state to inactive region is employed. This method isn't described in this paper.

The constraint neurons not to make the auxiliary neuron of the proposed model active except at the saddle point must be defined. For the i-th neuron, one is used to restrain the state change to the outside of the boundary from the point (1, 0) to the point $(x_{0i}, 1)$. This constraint condition is given by,

5

$$\frac{1}{\sqrt{1+(x_{0i}-1)^2}}\, x_i + \frac{x_{0i}-1}{\sqrt{1+(x_{0i}-1)^2}}\, u + \frac{1}{\sqrt{1+(x_{0i}-1)^2}} \leq 0. \qquad (10)$$

Another is used to restrain the state change to the outside of the boundary from the point (0, 0) to the point $(x_{0i}, 1)$. This constraint condition become as follows.

6

$$-\frac{1}{\sqrt{1+x_{0i}^{\,2}}}\, x_i + \frac{x_{0i}}{\sqrt{1+x_{0i}^{\,2}}}\, u \leq 0. \qquad (14)$$

From these inequality, the constraint vectors $\boldsymbol{R}_k$ and the threshold r_k can be derived easily.

3.4 Addition of noise at the saddle point

When the state of the network is arrived at the saddle point, the state can't be changed any more because the increments Δx and Δu become 0 at the saddle point. Therefore, in order to move to the less energy direction, the small noise must be added to the proposed network. Then in our simulation, when the activation of the auxiliary neuron become more than 0.95, the small noise ($|n|=0.3$) is added.

3.5 Detection of the least energy state

By repeating the state change described before, the proposed network can find the state on the u=0 plane which has less energy than the saddle point. Therefore, the least energy state of the original network can be found at last by decreasing gradually the energy level of the saddle point. However, this operation takes a lot of CPU time. Therefore, the method that the energy level of the saddle point is set to the same level as that of the state which have been found previously is employed. In this method, the saddle point memorizes the found energy level and behaves like a comparator.

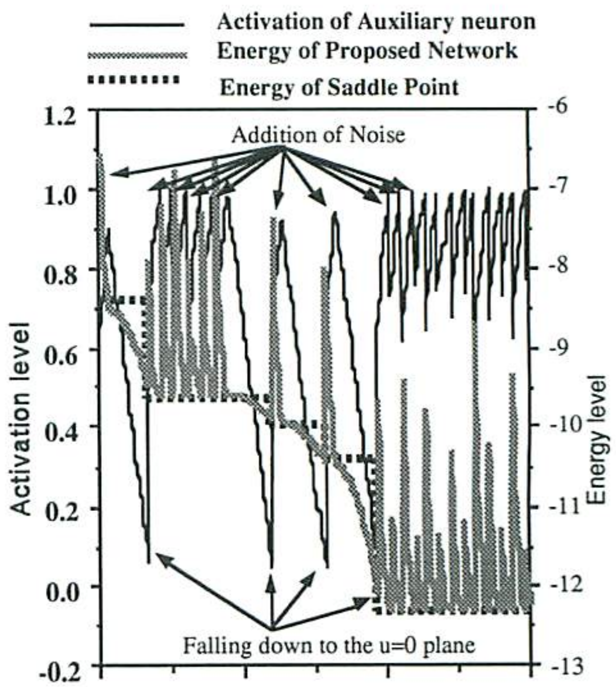

Fig.3 Activation of Auxiliary neuron and Energy of Proposed network while Network Running

4 SIMULATION

The proposed model was applied to the 5 cities TSP problem. This problem is to find the route which satisfies the following 3 conditions.

1 Salesman is not permitted to pass more than 2 cities at a same time

2 Salesman is not permitted to pass the same cities more than twice.

3 The total distance of the path must be the least path of the considerable paths.

The conditions 1 and 2 were introduced to the proposed network as the constraint neuron. The third condition was defined as the cost function. If the idea that salesman visits the x-th cities in the i-th turn is related to the neuron of which activation is v_{xi}, the cost function becomes as follows.

7

$$D(v_{xi}) = \sum_x \sum_i \sum_y \sum_j d_{xy}(\delta(i+1,j)+\delta(i-1,j))\, v_{xi} v_{yj} - \alpha\,(v_{xi}^{\,2} - v_{xi}), \qquad (10)$$

d_{xy}: the distance between the x-th city and the y-th city,

$\delta(i,j)$: Kronecker's delta(1, if x=0, 0, otherwise),

α : constant value.

where v_{xi} takes 0 or 1 value. This cost function is a quadratic form and can be replace to the original network energy function Eq.(2). Besides, if v_{xi} is 0 or 1, the addition of $v_{xi}^{\,2} - v_{xi}$ to this equation does not change the total distance $D(v_{xi})$. Therefore, in order to make this energy function upper-convex, this term was introduced.

This model was simulated by computer. The position of citites were given by the random function. Its x and y coordinates were restricted from 0 to 1. By changing the initial value for the random function, 10

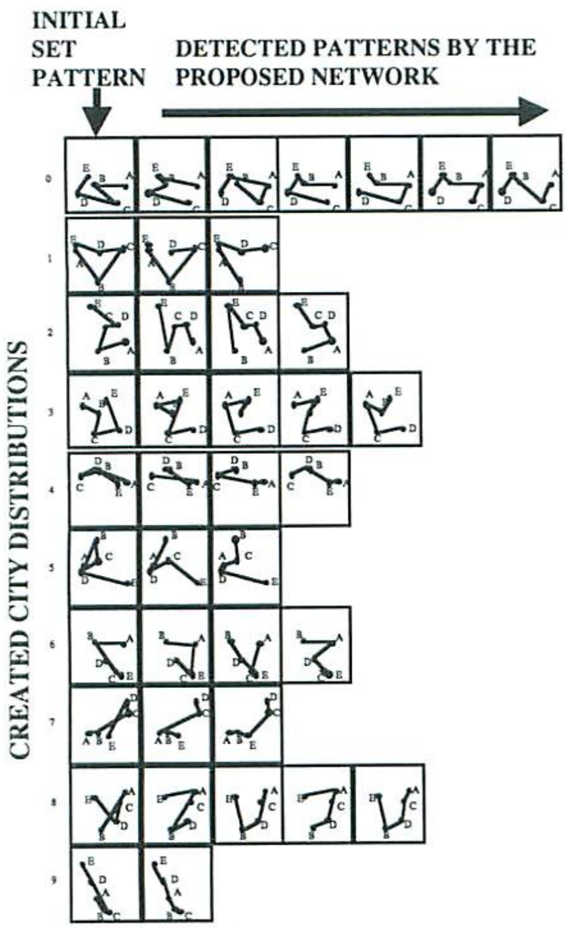

Fig. 4 The Detected Traveling Pattern By The Proposed Network

types of problems were made as shown in Fig.4.

Figure 3 shows the activation of the auxiliary neuron, u and the energy level of the proposed network while the network running for TSP problem made by the initial value 0. The points at which the activation of the auxiliary neuron is falling down shows that the proposed network is searching the descent energy path to the u=0 plane using noise. When the network found the descent path, the u value fell down to 0 and the energy level also fell down one step. The energy step is shown by the dot line. In this case, the energy level fell down 6 times though Fig.3 shows only 3 times falling down. After the 6-th falling down, the network could not escape from the saddle point and the processing was terminated.

Figure ~~9~~ 4 shows the results which was found by the proposed network for 10 TSP problems. In the row of figures, all the paths found until the network detected the least path are revealed. The number of the found paths was different by each problem. Besides, it was unpredictable what path the network may find on the process. However the proposed network could find at last the least path for each problem.

5 DISCUSSION

Now, we are researching the shape recognition model. TSP problem is not directly related to our purpose. However, the method to search the least energy state can be applied to the shape recognition. For example, human can recognize the shape irrelevantly to position, rotation and deformation. This shows that in human brain, the input image through eyes correctly correspond to the memorized image by neural network. This mechanism have not been know in detail. We tried to match the image patterns from camera with the simplified memorized patterns by Hopfield network. However, because of local minimum , they could been matched correctly. Then, the proposed network was required.

For checking this model using the real image data, the network needs a lot of neurons and CPU time. Then, we applied this model to TSP problem which can simplify the problem. As the results, the proposed model is found to detect the global minimum of the energy function but to have some problems.

This model took about a hour for one solution using SUN-work station. Besides, it needed more than 2 times constraint neurons of the neurons in the original network to control the network behavior. The calculation using the current computer will take a lot of time and memory in order to apply the real application. It is unavailable for the real application.

However we think that this model is available for analogue device and that if such the device are realized, the solution time will be improved. Neuron is essentially analogue. Therefore, the behavior of it isn't thought to be restrict in digital value. We don't know that the analogue devi~~s~~ce such as the proposed network can be realized. We hope any advice from the viewpoints of an analogue device.

6 CONCLUSION

The neural network model to escape local minimum is proposed and applied to 5 cities TSP problem. This model is found to takes many time but to find the global minimum definitely. We intend this model to be applied to the shape pattern recognition.

7 REFERENCE

[1] M.Grotschel, O.Holland: Solution of Large-Scale Symmetric Travelling Salesman Problem; Mathematical Programming, Vol.51, pp.141～202(1991)

[2] M.W.Padberg and G.Rinaldi: A Branch-and-Cut Algorithm for the Resolution of Large Scale Symmetric Traveling Salesman Problems; SIAM Review, Vol.33, pp.60～100(1991)

[3] Hopfield, J.J. and D.Tank: Neural computation of decisions in optimization problems; Biol. Cybernetics, Vol.52, pp. 141-152(1985)

[4] J. McClelland, D.E.Rumelhart: Parallel Distributed Processing; The MIT Press

[5] E. Aarts, J.Korst: Simulated Annealing and Boltzmann Machines, A Stochastic Approach to Combinatorial Optimization and Neural Computing; John Wiley & Sons Ltd, Tiptree Essex GB, (1989)

[6] H.Nozawa:CHAOS 2,3 (1992) 377.

[7] J.E.Dayhoff: Neural Network Architectures; Van Nostrand Reinhold

[8] K.Nakano: An Introduction to Neurocomputing; Corona Publishing Co. LTD.