Solving the Capacitated Vehicle Routing Problem by Columnar Competitive Model

Jingyu Li
Geography Department, Dezhou University, Dezhou, Shandong 253023, China
E-mail: jingyulidzu@163.com

***Abstract*—The vehicle routing problem (VRP) concerns the transport of items between depots and customers by means of a fleet of vehicles. The capacitated vehicle routing problem (CVRP) is the basic version of the VRP. A columnar competitive model (CCM) of neural networks incorporates with a winner-take-all learning rule is employed to solve the CVRP. Stability condition of CCM for CVRP is exploited by mathematical analysis. Parameters settings of the network for guaranteeing the network converges to valid solutions are discussed in detail. Simulations are carried out to illustrate the performance of the columnar competitive model.**

***Index Terms*—columnar competitive model (CCM), capacitated vehicle routing problem (CVRP), winner-take-all**

I. Introduction

The vehicle routing problem (VRP) concerns the transport of items between depots and customers by means of a fleet of vehicles. Examples of VRPs are: mail delivery, school bus routing, solid waste collection, milk delivery, dial-a-ride systems, heating oil distribution, parcel pick-up and delivery, and many others.

Finding optimal routes for a fleet of vehicles performing assigned tasks on a number of spatially distributed customers can be formulated as a combinatorial optimization problem: the vehicle routing problem. A solution of this problem is the best route serving all customers using a fleet of vehicles, respecting all operational constraints, such as vehicle capacity and the driver's maximum working time, and minimizing the total transportation cost.

The capacitated vehicle routing problem (CVRP) is the basic version of the VRP. The name derives from the constraint of having vehicles with limited capacity. Customer demands are deterministic and known in advance. Deliveries cannot be split, that is, an order cannot be served using two or more vehicles. The vehicle fleet is homogeneous and there is only one depot. The objective is to minimize the total travel cost, usually expressed as the traveled distance required to serve all customers. The CVRP is NP-hard [1] and the size of the problems which can be solved exactly in a reasonable time is up to 50 customers, using the branch-and-bound, branch-and-cut, and set-covering approaches.

II. The Mathematical Model of CVRP

In general, (CVRP) meets the following requirements:

- Customers locate in distribution area, and a single customer demand is less than the capacity of one vehicle.
- Each customer can get the timely delivery service, and each customer is only visited by one vehicle only one time.
- Every vehicle can only service one route, the distribution vehicle is initial and terminated in only one depot.
- On each distribution lines, the total demand of each customer is no more than the capacity of one vehicle.

The depot position, the customer position and road conditions are known. The considered costs include the running costs of the vehicles and time, distance and related expenses. In full considering the requirements of the CVRP and optimized object, the mathematical model of CVRP is established.

CVRP assume that there is one depot and n customers. The depot is numbered 0 and customer is numbered 1, 2…, n. The transportation cost between i customer and j customer is d_{ij}. The demand of i customer is q_i and the maximum load of the vehicles is T. The number of vehicle is m.

First, the following variables are defined:

$$y_{ik} = \begin{cases} 1, & If\ the\ delivery\ of\ customer \\ & i\ is\ finished\ by\ vehice\ k \\ 0, & otherwise \end{cases} \tag{1}$$

$$x_{ijk} = \begin{cases} 1, & If\ the\ vehicle\ k\ drives\ from\ customer \\ & i\ to\ customer\ j \\ 0, & otherwise \end{cases} \tag{2}$$

The objective function,

$$\min Z = \sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{k=1}^{m} d_{ij}x_{ijk} \tag{3}$$

$$\text{s.t. } \sum_{k=1}^{m} y_{ik} = 1, i = 1,2,...,n \tag{4}$$

Manuscript received January 3, 2014; revised March 27, 2014.

doi: 10.12720/jtle.2.1.1-6

$$\sum_{i=0}^{n}\sum_{k=1}^{m} x_{ijk} = 1,\ j = 0,1,\ldots,n \tag{5}$$

$$\sum_{j=1}^{n}\sum_{k=1}^{m} x_{ijk} = 1, i = 0,1,\ldots,n \tag{6}$$

$$\sum_{i} q_i y_{ik} \le T \tag{7}$$

$$\sum_{i=1}^{n}\sum_{k=1}^{m} x_{0ik} = \sum_{j=1}^{n}\sum_{k=1}^{m} x_{j0k} \tag{8}$$

Equation (4), (5) and (6) guarantee that each customer is only visited by one vehicle only one time; Equation (7) ensures that the total demand of each customer is no more than the capacity of one vehicle on each distribution lines; Equation (8) guarantees the distribution vehicle is initial and terminated in only one depot.

III. The Hopfield Neural Networks and Columnar Competitive Model (CCM)

A Hopfield network is a form of recurrent artificial neural network invented by John Hopfield. Hopfield nets serve as content-addressable memory systems with binary threshold nodes. They are guaranteed to converge to a local minimum, but convergence to a false pattern (wrong local minimum) rather than the stored pattern (expected local minimum) can occur.

There has been an increasing interest in applying the Hopfield neural networks to combinatorial optimization problems, since the original work of Hopfield and Tank [2]. Several methods have been proposed to ensure the network converges to valid states. Aiyer *et al*. [3] have theoretically explained the dynamics of network for traveling salesman problems by analyzing the eigen values of the connection matrix. Abe [4] has shown the theoretical relationship between network parameters and solution quality based on the stability conditions of feasible solutions. Chaotic neural network provides another promising approach to solve those problems due to its global search ability and remarkable improvement with less local minima, see in [5]–[8]. Peng *et al*. [9] suggested the local minimum escape (LME) algorithm, which improves the local minimum of CHN by combining the network disturbing technique with the Hopfield network's local minimum searching property. Otherwise, many papers have discussed efficient mapping approaches. Talavan and Yanez [10] presented a procedure for parameters settings based on the stability conditions of the network. Cooper et al. [11] developed the higher-order neural networks (HONN) to solve TSP and study the stability conditions of valid solutions. Brandt *et al*. [12] presented a modified Lyapunov function for mapping the TSP problem. All of those works are noteworthy for the solving of TSP.

To solve CVRP efficiently, a new neural network is employed, which has a similar structure as Hopfield network, but it obeys a different updating rule: the columnar competitive model (CCM), which incorporates winner-takes-all (WTA) in column-wise.

Competitive learning by winner-takes-all (WTA) has been recognized to play an important role in many areas of computational neural networks, such as feature discovery and pattern classification [13] - [15]. Nevertheless, the potential of WTA as a means of eliminating all spurious states is seen due to its intrinsic competitive nature that can elegantly reduce the number of penalty terms, and hence the constraints of the network for optimization. The WTA mechanism can be described as: given a set of n neurons, the input to each neuron is calculated and the neuron with the maximum input value is declared the winner. The winner's output is set to '1' while the remaining neurons will have their values set to '0'.

In the work of Hopfield [1], the form of an energy function is

$$E = -\frac{1}{2}V^T W V - (i^b)^T V \tag{9}$$

It can be minimized by the continuous-time neural network with the parameters: $W = (W_{i,j})_{n\times n}$ is the connection matrix, n is the number of neurons, $V = (V_{x,i})_{n\times n}$ represents the output sate of the neuron (x, i), and i^b is the vector form of bias.

The energy function of CCM for CVRP can be written as

$$E = \frac{A}{2}\sum_{x}\sum_{i}\left(v_{x,i}\sum_{j\neq i} v_{x,i}\right) + \frac{B}{2}\sum_{i}^{m}\left(S_i - \bar{S}\right)^2 + \frac{1}{2}\sum_{x}\sum_{y\neq x}\sum_{i} d_{xy} v_{xi}\left(v_{y,i+1} + v_{y,i-1}\right) \tag{10}$$

where A > 0, B > 0 are scaling parameters, d_{xy} is the distance between customer x and y, $\bar{S} = \frac{n}{m}$ is the average number of customers for m tours, S_i is the number of cities that have been visited by vehicle i.

Let m_i (i = 1,. . . , m-1) be the index of the virtual customer in the whole customer's sequence which composed by m vehicle's tour concatenated end by end, and set $m_0 = 0$, $m_m = n + m - 1$, n is the total number of customers.

Comparing (9) with (10), the connection matrix and input basis are computed as

$$W_{xi,yi} = -\left\{A\delta_{xy}(1-\delta_{ij}) + d_{xy}(\delta_{i,j+1} + \delta_{i,j-1}) + B\theta_j\right\}$$

$$i^b = \frac{n}{m}$$

where

$$\theta_j = \begin{cases} 1, & if\ m_{i'-1} \le j < m_i \\ 0, & otherwise \end{cases}$$

Then the input to neuron (x, i) is calculated as

$$Net_{x,i} = \sum_y \sum_i (W_{xi,yj} v_{yj}) + i^b$$

$$= -\sum_y d_{xy}(v_{y,i-1} + v_{y,i+1}) - A\sum_{j \ne i} v_{x,j} - B\sum_y \sum_{j=m_{i'-1}}^{m_{i'}-1} v_{y,j} + \frac{n}{m}$$

$$= -\sum_y d_{xy}(v_{y,i-1} + v_{y,i+1}) - A\sum_{j \ne i} v_{x,j} - BS_{i'} + \frac{n}{m} \qquad (11)$$

The columnar competitive model based on winner-take-all (WTA) leaning rule, the neurons compete with others in each column, and the winner is with the largest input. The updating rule of outputs is given by

$$v_{x,i}^{t+1} = \begin{cases} 1, & \text{if } Net_{x,i}^t = \max\{Net_{1,i}^t, Net_{1,i}^t, ..., Net_{N+m-1,i}^t\} \\ 0, & \text{otherwise} \end{cases}$$

For CCM, $v_{x,i}$ is updated by the above WTA rule. The whole algorithm is summarized as follows:

The CCM algorithm
1 Initialize the network, with each neuron having a small initial value $v_{x,i}$. A small random noise is added to break the initial network symmetry. Compute the W matrix ;
2 Select a column (e.g., the first column). Compute the input $Net_{x,i}$ of each neuron in that column;
3 Apply WTA , and update the output state of the neurons in that column.;
4 Go to the next column, preferably the one immediately on the right for the convenience of computation. Repeat step 3 until the last column in the network is done. This constitutes the first epoch.
5 Go to step 2 until the network converges (i.e., the states of the network stop changes).

IV. The Parameters Settings for the CCM when it be Applied to CVRP

For the energy function of CCM, the critical value of the penalty-term scaling parameter A and B play a predominant role in ensuring its convergence and driving the network to converge to valid states.

Consider the p-column of neuron outputs states matrix, suppose row b is an all-zero row and row a is not all-zero. According to Eq. (8), the input to neuron (a,p) and (b,p) is computed as

$$Net_{a,p} = -\sum_y d_{ay}(v_{y,p-1} + v_{y,p+1}) - A\sum_{j \ne p} v_{a,j} - BS_{p'} + \frac{n}{m}$$

$$Net_{b,p} = -\sum_y d_{by}(v_{y,p-1} + v_{y,p+1}) - A\sum_{j \ne p} v_{b,j} - BS_{p'} + \frac{n}{m}$$

where $0 < S_{p'} < n$, Suppose row a contains l "1" (1 <= l <= n + m -1), then

$$Net_{b,p} = -\sum_y d_{by}(v_{y,p-1} + v_{y,p+1}) - BS_{p'} + \frac{n}{m}$$

$$> -\sum_y d_{by}(v_{y,p-1} + v_{y,p+1}) - Bn^2 + \frac{n}{m}$$

$$Net_{a,p} = -A(l-1) - \sum_y d_{ay}(v_{y,p-1} + v_{y,p+1}) - BS_{a'} + \frac{n}{m}$$

$$< -A(l-1) - \sum_y d_{ay}(v_{y,p-1} + v_{y,p+1}) + \frac{n}{m}$$

Let $A - Bn^2 > 2d_{\max} - d_{\min}$, where *dmax* and *dmin* is the maximum and the minimum distance, respectively. Then the CCM by is always convergent to valid states.

It is clear that only one neuron's output in per column be set to '1' under WTA updating rule. Assume the network reaches the following state after some updating:

$$v = \begin{bmatrix} ... & ... & ... & ... & ... \\ 0 & v_{b,p} & 0 & 0 & 0 \\ 1 & v_{s,p} & 0 & ... & 0 \\ 0 & v_{t,p} & 1 & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & v_{n,p} & 0 & ... & 1 \end{bmatrix}$$

The input to each neuron in the pth column is calculated as

$$Net_{b,p} > -(d_{bt} + d_{bs}) - Bn^2 + \frac{n}{m}$$

$$Net_{s,p} < -(A + d_{st}) + \frac{n}{m}$$

$$Net_{t,p} < -(A + d_{st}) + \frac{n}{m}$$

$$Net_{t,p} < -(A + d_{sn} + d_{tn}) + \frac{n}{m}$$

To ensure the neuron's outputs reach the valid solution in the next updating, the neuron $v_{b,p}$, should be the winner, since all the other neurons in row b is zero. In the other word, the input of the neuron $v_{b,p}$ should be the maximum one of all the inputs in pth column, $Net_{b,p} > Net_{a,p}, a \ne b$. Therefore it is ensured by

$$-(d_{bt} + d_{bs}) - Bn^2 + \frac{n}{m} > -(A + d_{st}) + \frac{n}{m}$$

$$-(d_{bt} + d_{bs}) - Bn^2 + \frac{n}{m} > -(A + d_{sn} + d_{tn}) + \frac{n}{m}$$

It is well known that $d_{st} < d_{sn} + d_{tn}$. Then it follows that $A - Bn^2 > d_{bt} - d_{bs} - d_{st}$, which can be ensured by $A - Bn^2 > 2d_{\max} - d_{\min}$. The neuron outputs always escape from the invalid states.

V. To Investigate the Dynamical Stability of the CCM

It is known that the stability of the original Hopfield networks is guaranteed by the well-known Lyapunov energy function. However, the dynamics of the CCM is so different from the Hopfield network, thus the stability of the CCM needs to be investigated.

To investigate the dynamical stability of the CCM, a $n \times n$ network is considered. After the *n*-th WTA updating, the network would have reached the state with only one '1' per column, but may have more than one '1' per row. Suppose v_t and v_{t+1} are two states before and after WTA updating respectively. Consider *p*-th column, and let neuron (*a*, *p*) be the only active neuron before updating, i.e., $v_{a,p}^t = 1$ and $v_{i,p}^t = 0$, $\forall i = a$. After updating, let neuron (*b*, *p*) be the winning neuron, i.e., $v_{b,p}^{t+1} = 1$, $v_{i,p}^{t+1} = 0$, $\forall i = b$.

The energy function can be broken into three terms E_P, E_q, E_0, that is, $E = E_p + E_q + E_0$. E_P stands for the energy of the columns p -1, p and p + 1 of the rows a and b. E_q stands for the energy of the groups. E_0 stands for the energy of the rest columns and rows. Then E_P is computed by

$$E_p = \frac{A}{2}\sum_i v_{a,i}\sum_j v_{b,j} + \sum_x\sum_y d_{xy} v_{x,p}(v_{y,p+1} + v_{y,p-1})$$

E_0 is computed as

$$\begin{aligned} E_0 = {} & \frac{A}{2}\sum_{x\neq a,b}\sum_i \left(v_{x,i}\sum_{j\neq i} v_{x,j} \right) \\ & + \frac{1}{2}\sum_x\sum_y\sum_{i\neq p-1,p,p+1} d_{xy} v_{x,i}(v_{y,i-1} + v_{y,i+1}) \\ & + \frac{1}{2}\sum_x\sum_y d_{xy} v_{x,p-1} v_{x,p-2} \\ & + \frac{1}{2}\sum_x\sum_y d_{xy} v_{x,p+1} v_{x,p+2} \end{aligned}$$

And E_q is calculated as

$$E_q = \frac{B}{2}\sum_{i=1}^{m} S_i^2 - \frac{n(n+m-1)}{m} + \frac{Bn^2}{2m}$$

To investigate how *E* changes under the WTA learning rule, the following two cases are considered.

Case 1: (a,p) and (b,p) are both not group neuron

In this case 0 < a, b < n, it can be seen that only E_P will be affected by the state of column p. E_p^t and E_p^{t+1} is computed by

$$E_p^t = -\frac{A}{2} l(l-1) - \sum_y d_{ay}\left(v_{y,p-1} + v_{y,p+1}\right)$$

$$E_p^{t+1} = -\frac{A}{2} l(l-1)(l-2) - \sum_y d_{by}\left(v_{y,p-1} + v_{y,p+1}\right)$$

At the same time, the input to neuron (a,p) and (b,p) before updating are computed as follows:

$$Net_{a,p}^t = -A(l-1) - \sum_y d_{ay}(v_{y,p-1} + v_{y,p+1}) - BS_{p'} + \frac{n}{m}$$

$$Net_{b,p}^t = -\sum_y d_{by}(v_{y,p-1} + v_{y,p+1}) - BS_{p'} + \frac{n}{m}$$

According to the discussion of the above subsection, when $A - Bn^2 > 2d_{\max} - d_{\min}$ is guaranteed, then $Net_{b,p}^t > Net_{a,p}^t (a \neq b)$ can be ensured. This implies that $E_p^{t+1} - E_p^t < 0$, $E^{t+1} - E^t < 0$.

Case 2: (a,p) is a group neuron while (b,p) is not

In this case n < a < n + m while 0 < b < n, it can be concluded that not only E_P but also E_q would be changed before and after the WTA updating rule in pth column. (a, p) is group neuron, it can be active before updating while non-active after updating. This implies that two connected groups which be distinguished by neuron (a, p) before updating merge into one group after updating. Suppose s_g^t and s_h^t represent the customer's number of those two connected groups before updating, and s_p^{t+1} stands for the customer's number of this merged group after updating. Then, $s_g^t + s_h^t = s_p^{t+1}$. E_q^t and E_q^{t+1} are computed as

$$\begin{aligned} E_q^t = {} & \frac{B}{2}(s_g^t)^2 + \frac{B}{2}(s_h^t)^2 \\ & + \frac{B}{2}\sum_{i=,i\neq g,h} S_i^2 - \frac{n(n+m-1)}{m} + \frac{Bn^2}{2m} \end{aligned}$$

$$E_q^t = \frac{B}{2}(s_p^{t+1})^2 + \frac{B}{2}\sum_{i=,i\neq g,h}^{m} S_i^2 - \frac{n(n+m-1)}{m} + \frac{Bn^2}{2m}$$

The change of E is

$$\begin{aligned} E_p^{t+1} + E_q^{t+1} - E_p^t - E_q^t = {} & -A(l-1) \\ & + \sum_y d_{by}(v_{y,p+1} + v_{y,p-1}) + Bs_g^t s_h^t \\ & - \sum_y d_{ay}(v_{y,p+1} + v_{y,p-1}) \end{aligned}$$

When $A - Bn^2 > 2d_{\max} - d_{\min}$, $Net_{b,p}^t > Net_{a,p}^t$ can be ensured, it implies that

$$-A(l-1)+\sum_y d_{by}(v_{y,p+1}+v_{y,p-1})+Bs_g^t s_h^t$$
$$-\sum_y d_{ay}(v_{y,p+1}+v_{y,p-1})<-Bn^2$$

Thus $-Bn^2+Bs_g^t s_h^t<-Bns_h^t+Bs_g^t s_h^t<0$, and it also implies that $E^{t+1}-E^t<0$.

VI. The Neural Network Model Presented to Solve CVRP

It is clear that there are m sub-tours in a valid solution of the n-customers and m-loop CVRP problems, and each sub-tour starts with the starting city. To map this problem into the network, an appropriate representation scheme of neurons should be required. It is well known, a n*n square array has been used in n-city TSP problem by Hopfield, but it is not enough for CVRP problem.

Thus, m -1 customers are added to the network, and those customers are called as the virtual cities. The virtual customers have the same connections and weights as the starting depot. The location information of a virtual customer is specified by the output state of m -1 neurons in the representation scheme. Those neurons are defined as the group neurons. Hence, m loops are distinguished easily by those virtual customers using a (n + m -1) * (n + m-1) square matrix. For example, a 4- customer and 2-loop problem, an original feasible solution and the new one after adding the virtual costumer as shown in Fig. 1. It needs (5 + 2 - 1) * (5 + 2 -1) = 36 neurons to represent the neurons's output states.

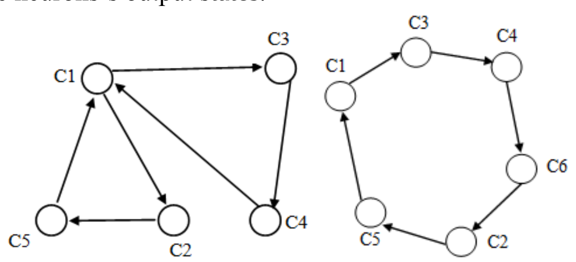

Figure 1. The feasible solution and the tour with the virtual costumer.

In Fig. 1, the left is the feasible solution tour. The right is the tour after adding the virtual depot. C1 is the depot, and F is the virtual depot.

The following output states of neurons represent a multi-tour with 2 tours. C1 is the starting depot. C3 is the second and C4 is the third of the first tour. C6 is the virtual depot which represent depot C1. The second tour starting from C6(C1), visited C2, C5 in sequence. To sum up, the first tour is C1 _ C3 _ C4 _ C6(C1) and the second is C6(C1) _ C2_ C5 _ C1(C6).

	1	2	3	4	5	6
C1	1	0	0	0	0	0
C2	0	0	0	0	1	0
C3	0	1	0	0	0	0
C4	0	0	1	0	0	0
C5	0	0	0	0	0	1
C6	0	0	0	1	0	0

VII. The Simulation Experiment

Assume 19 customers are randomly distributed in the fields of square area whose edge is 10 kilometers. The depot is located in the centre of square area, whose coordinates are (0, 0). The demand of every customer is generated by the computer randomly, and the vehicle load is 9 ton. Customer Number, customer coordinates and demand are shown in Table I.

TABLE I. Number, Coordinates and Demand of Customers

customer	N01	N02	N03	N04	N05	N06	N07
coordinates	(1,-1)	(1,-2)	(-4,-1)	(-4,0)	(1,3)	(-4,-4)	(-2,-2)
demand	1.7	3.0	2.5	1.0	0.6	0.8	2.0
customer	N08	N09	N10	N11	N12	N13	N14
coordinates	(0,3)	(0,-1)	(3,-1)	(-1,-1)	(-3,2)	(1,-4)	(2,1)
demand	1.8	1.5	1.5	0.1	3.1	2.2	0.5
customer	N15	N16	N17	N18	N19		
coordinates	(2,-1)	(1,-3)	(2,0)	(-3,0)	(3,4)		
demand	0.7	2.0	1.9	2.4	0.2		

With the CCM, the best solution is four routes:

route1: 0→12→0;

route2: 0→9→13→16→2→0;

route3: 0→18→4→3→6→7→11→0;

route4: 0→8→5→19→14→3→10→15→1→0.

VIII. Conclusion

In the research, a new columnar competitive model (CCM) incorporating the WTA learning rule has been proposed for solving CVRP which is the basic version of the VRP.

The stability condition of CCM for CVRP was exploited. According to the theoretical analysis, the critical values of the network parameters were found. The simulation result shown that WTA updating rule makes CCM an efficient and fast computational model for solving CVRP.

For time limit, the algorithm is just tested on simulation experiments. In the future, more tests will be carried out to check the performance of the algorithm. Moreover, there is still a big room for improvement of the algorithm in the future. Also, the proposed CCM will be applied to other problems.

Acknowledgment

This work is supported by Shandong province office of education (No.S07WZ35)

References

[1] M. Labbé, G. Laporte, and H. Mercure, "Capacitated vehicle routing on trees," *Operations Research*, vol. 39, no. 4, pp. 616–622, 1991.

[2] J. J. Hopfield and D. W. Tank, "'Neural' computation of decisions in optimization problems," *Biological Cybernetics*, vol. 52, no. 3, pp. 141–52, 1985.

[3] S. V. B. Aiyer, M. Niranjan, and F. Fallside, "A theoretical investigation into the performance of the hopfield model," *IEEE Trans Neural Networks*, vol. 1, pp. 204–215, June 1990.

[4] S. Abe, "Global convergence and suppression of spurious states of the Hopfield neural networks," *TEEE Trans Circ Syst*, vol. 40, no. 4, pp. 246–257, 1993.

[5] 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.

[6] 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.

[7] M. Hasegawa, T. Lkeguchi, and K. Aihara, "Solving large scale traveling problems by chaotic neurodynamics," *Neural Networks*, vol. 15, pp. 271–385, 2002.

[8] L. Wang and K. Smith, "On chaotic simulated annealing," *IEEE Trans Neural Networks*, vol. 9, no. 4, pp. 716–718, 1998.

[9] M. Peng, K. Narendra, and A. Gupta, "An investigation into the improvement of local minimum of the Hopfield networks," *Neural Networks*, vol. 9, pp. 1241–53, 1996.

[10] P. M. Talavan and J. Yanez, "Parameter setting of the Hopfield network applied to TSP," *Neural Networks*, vol. 15, pp. 363–73, 2002.

[11] B. Cooper, "Stability analysis of higher-order neural networks for solving combinatorial optimization problems," *Int J Neural Syst*, vol. 12, no. 3, pp. 177–186, 2002.

[12] R. D. Brandt, Y. Wang, A. J. Laub, and S. K. Mitra, "Alternative networks for solving the traveling salesman problem and the list-matching problem," in *Proc. International Joint Conference on Neural Networks*, vol. 2, 1998, pp. 333–340.

[13] D. E. Rumelhart and D. Zipser, "Feature discovery by competitive learning," *Cognitive Science*, vol. 9, pp. 75-112, 1985.

[14] L. Wang, "On competitive learning," *IEEE Trans Neural Networks*, vol. 8, no. 5, pp. 1214–1216, 1997.

[15] J. P. F. Sum, C. S. Leung, P. K. S. Tam, G. H. Young, W. K. Kan, and L. W. Chan, "Analysis for a class of winner-take-all model," *IEEE Trans Neural Networks*, vol. 10, no. 1, pp. 64–70, 1999.

Jingyu Li Dezhou University.
Jingyu Li, Born Date: October, 1991, Nationality: Shandong Province, China. bachelor degree of Engineering is proceeded from DEZHOU University, Shandong, China.
His research Interests is at GIS Applications, Optimization theory and algorithms, Data Mining, VRP.
Jingyu Li is a assistant research fellow of Geography Department, Dezhou University, Dezhou, Shandong Province, China.

