A Neural Network Based on Algorithm for Traveling Salesman Problem

Cui Zhang[1], Xiaofei Li[2]

1. School of Electrical and Automation Engineering, Liaoning Institute of Science and Technology, Benxi 117020, China.

2. MCC Coke Resistance (Dalian) Engineering Technology Co., Ltd, Dalian 117020, China.

Abstract: This paper construct a neural network based algorithm to solve the traveling salesman problem. The performance of the algorithm is evaluated through simulating 100 randomly generated instances of the 10-city traveling salesman problems. The performance of the proposed learning method on these test problems is very satisfactory in terms of solution quality.

Keywords: Binary Neuron Model; Combinatorial Optimization Problems; Traveling Salesman Problem

1. Introduction

The idea of using neural networks to provide solutions to difficult NP-hard optimization problems [1], [2] originated in 1985 when Hopfield and Tank demonstrated that the traveling salesman problem (TSP) could be solved using a Hopfield neural network [3], Since Hopfield and Tank's work, there has been growing interest in the Hopfield network because of its advantages over other approaches for solving optimization problems. The advantages include massive parallelism, convenient hardware implementation of the neural network architecture, and a common approach for solving various optimization problems.

The Hopfield neural networks [3] constitute an important avenue. These networks contain many simple computing elements (or artificial neurons) which cooperatively traverse the energy surface defined by E(v) to find a local or global minimum. The simplicity of the neurons makes it promising to build them in large numbers to achieve high computing speeds by way of massive parallelism. Also the inherent parallelism in the neural network provides a promising alternative for solving these problems. Almost every type of combinatorial optimization problem has tackled by neural network . The first neural network representation for solving a combinatorial optimization problems was introduced by Hopfield and Tank [3]. Unfortunately, the technique, which requires minimization of an energy function containing several terms and parameters, was shown that it often failed to converge to valid solutions and when it converged, the obtained solution was often far from the optimal solution by Wilson and Pawley in 1988. Since then, various modifications have been proposed to improve the convergence of the Hopfield network.

Focusing on the energy function, it was showed that the convergence of the Hopfield network could be improved by modifying the energy function. Protzel et al. studied Brandt et al.'s formulation with different parameters. They found that despite the improved convergence of the modified versions of the Hopfield network, the network might not converge to solutions with good quality. Takefuji et al. showed that the decay term in the Hopfield neural network increases the energy function under some conditions . They modified the motion equation in order to guarantee the local minimum convergence. However, with the Hopfield neural network, the state of system is forced to converge to the local minimum. In other words, the neural network cannot always find the optimum solution. Therefore, several neuron models and heuristics such as hysteresis binary neuron model, neuron filter, hill-climbing term and omega function , Lagrange relaxation and pots spin have been proposed to improve the convergence of the networks. In addition, some researchers tried to modify the internal dynamics of the network to improve the search quality of solution of the Hopfield network.

Despite the improvement of the performance of the Hopfield network over more than a decade, this model still has some basic problems. One of the problems is that the performance of the Hopfield network is very sensitive to weights (parameter) for constraint term and cost term in energy function of network. There is no rule of thumb to find out an appropriate parameter set. Instead, trial-and-error has to be applied.

We have proposed a saturation binary neuron model to improve the performance of the Hopfield network. In this paper, we use saturation binary neuron model to construct a Hopfield-type neural network for efficiently solving traveling salesman problems. In the proposed method, once the neuron is in excitatory state, its input potential is in positive saturation where the input potential can only be reduced but cannot be increased, and once the neuron is in inhibitory state, then its input potential is in negative saturation where the input potential can only be increased but cannot be reduced. Using the saturation binary neuron model, a saturation binary neural network is constructed to solve the traveling salesman problems.

2. Traveling salesman problems

The traveling salesman problem is one of the most famous problems in combinatorial optimization. Nowadays, it plays a very important role in the development and testing of new optimization techniques. In this section, we apply the proposed learning method to the TSP.

The traveling salesman problem consist of finding the shortest closed path by which every city out of a set of N cities is visited once and only once. In the Hopfield neural network approach, a TSP instance is represented by an energy function including cost and constraint terms that reflect the objective of a solution. The objective of the constrain term is to find a valid tour, which requires that each city must be visited once and only once. The objective of the cost term is to find the shortest valid tour. For an N-city TSP problem, the network consists of $N \times N$ neurons and the neurons are fully inter-connected. The row index for a neuron represents the city. The column index represents the order of the city in the tour. Therefore, the constraint term of the energy function can be described as follow:

$$E_1 = \sum_{i=1}^{N}\left(\sum_{j=1}^{N} y_{ij} - 1\right)^2 + \sum_{l=1}^{N}\left(\sum_{k=1}^{N} y_{kl} - 1\right)^2 \quad (1)$$

Where i and k are row indices; j and l are column indices; yij is the state of #ij neuron. The first term of Eq.(15) enforce the constraint that no city can be visited more than once and the second term of Eq.(15) does not allow the salesman to visit two different cities at the same time.

The cost term of the energy function is given by:

$$E_2 = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1,k\neq i}^{N} d_{ik} y_{ij}(y_{k,i+1} + y_{k,i-1}) \quad (2)$$

Where dik is a measure of the distance between cities i and k. Thus, the total energy function of TSP can be described as follow:

$$E = A \cdot E_1 + B \cdot E_2 \quad (3)$$

Where A, B are parameter which are used as the weight for the constraint to ensure feasible solution and the weight for the optimization criterion, respectively.

3. Proposed Neuron Model

In order to improve the global convergence quality and shorten the convergence time, we have proposed a new neuron model called saturation binary neuron model.which consists of the following important ideas.

Once the neuron is in excitatory state (the output yi=1), then the input potential is assumed to be in positive saturation. In the positive saturation, the input potential xi can only be reduced but cannot be increased.

For the case of yi=1:

if $dx_i(t)/dt < 0$

$$x_i(t+1) = x_i(t) + \frac{dx_i(t)}{dt} \quad (4)$$

else

$$x_i(t+1) = x_i(t) \quad (5)$$

Once the neuron is in inhibitory state (the output yi=0), then the input potential is assumed to be in negative saturation. Then the input potential can only be increased but cannot be reduced.

For the case of yi=0

if $dx_i(t)/dt > 0$

$$x_i(t+1) = x_i(t) + \frac{dx_i(t)}{dt} \quad (6)$$

else

$$x_i(t+1) = x_i(t) \quad (7)$$

The saturation computation method uses the updating conditions (Eq.(4) - Eq.(7)) to update the input potential of each neuron. According to the updating conditions (Eq.(4) and Eq.(5)), when the output state of neuron is 0, its input potential can only be increased but cannot be decreased and once the input potential exceed the UTP, the output state of the neuron becomes 1 and the input potential will be in positive saturation. Thereafter the input potential will not be increased until the input potential falls into a negative saturation. Thus, different neuron may have different range for the input potential, and the same neuron may have different range for the input potential when the network is in different configuration. This kind of variation behavior of input potential in the saturation computation method can restrain the undesirable oscillation of network and the insensitivity to the input of neurons, and may result in better performance for the network. Using the above neuron model, we construct a Hopfield-type neural network called saturation binary neuron network. The following procedure describes the synchronous parallel algorithm using the saturation binary neuron network to solve a COP. Note that N is the number of neuron, targ_cost is the target total cost set by a user as an expected total cost and t_limit is maximum number of iteration step allowed by user.

1. Set t=0 and set targ_cost, t_limit, and other constants.
2. The initial value of xi for i=1,…,N is randomized.
3. Evaluate the current output yi(t) for i=1,…,N.
4. Terminate this procedure if targ_cost is reached.
5. Increment t by 1. if t> t_limit, terminate this procedure.
6. For i=1,…,N

a. Compute $dx_i(t)/dt$

b. Update xi(t+1), using the saturation binary neuron model (Eq.(4)~Eq.(7)).

7. Go to the step 3.

4. Simulation Results

In order to assess the effectiveness of the proposed learning method, extensive simulations were carried out over randomly generated instances of the 10-city traveling salesman problems on PC Station.

The first TSP instance that we tested was a randomly generated 10-city traveling salesman problem. From simulation, we found that when LTP and UTP are near zero or very large (LTP<-10 and UTP>25), the rate to find a valid solution is very low. Especially we can see that when LTP and UTP are zero, the network did not find a valid solution. The reason is that when LTP and UTP is zero, the proposed method is nearly as same as the time-independent computation method, and undesirable oscillation appears.

Many previous studies used only one data set (the one used by Hopfield and Tank in their original paper [3]) or a small number of data sets in their simulations. This may result in an unreliable conclusion when used in evaluating an algorithm. The reason is that the performance of an algorithm often depends on the location distribution of the cities in a data set. In order to reduce this effect and exactly evaluate the proposed learning method, we randomly generated 100 data sets of 10-city problems. For each data set, 100 runs with different initial input value of neuron were performed. In these 10000 runs, we found that the proposed learning method can find hundred percent valid solutions with short computation time, and the solution quality is very good.

Conclusion

We have proposed a neural network based algorithm. The proposed neuron network based algorithm is used to solve the traveling salesman problem. The simulation results show that the proposed neuron network based method is capable of finding 100 percent valid solution in a short time. Also, it can be seen that the proposed method is problem independent and can be used to solve other combinatorial optimization problems.

References

[1] P.W. Protzel, D. L. Plumbo, and M. K. Arras, “Performance and fault-tolerance of neural network for optimization,” IEEE Trans. Neural Networks, vol.4, pp.600-614, 1993.

[2] Y. Takefuji and K. C. Lee, “Artificial neural networks for four-coloring map problems and K-colorability problems,” IEEE Trans. Circuits Syst., vol.38, no. 3, pp.326-333, Mar. 1991.

[3] Y. Takefuji and K. C. Lee, “An artificial hysteresis binary neuron: A model suppressing the oscillatory behaviors of neural dynamics,” Biol. Cybern., vol.64, pp.353-356, 1991.