Incorporating Elastic Ring Into Potts Neural Network

Jiann-Ming Wu and Cheng-Yuan Liou
Department of Computer Science and Information Engineering
National Taiwan University, Taipei, Taiwan

Correspondence: Cheng-Yuan Liou, Department of Computer Science and Information Engineering, National Taiwan University, Taipei,Taiwan,10764, R.O.C. Tel. 8862-3630231 ext. 3229, FAX : 8862-3628167, email: cyliou@csie.ntu.edu.tw

Abstract

We devise a mechanism to include the elastic ring into the Potts neural encoding approach to solve the travelling salesman problem(TSP) with better performance. The obtained network is then named elastic Potts neural network (EPNN) which has much loosely coupled interconnections. Its function is close to the elastic ring method with different controlling mechanism for the ring path. The EPNN consumes much less computation in finding solution with the including of the geometrical abstraction of this elastic ring for the TSP. Based on averages of simulations, we find that the performance of EPNN is much better than that of the simmulated annealing method. The EPNN shows high stability in its solutions. The problem size can be extended easily to a 500-city TSP on a workstation.

1   Introduction

Two interesting analog approaches to the TSP problem, elastic ring method [2] and Hopfield and Tank neural network [3], are recently shown to be closely related by Simic [6] through the underlying theory of statistic mechanism and by Yuille in the work[8] on stereo vision models. By incorporating the elastic ring method into the Potts neural encoding, we further demonstrate one possible interactive mode of the two physics-spired models in this paper. As a result, we obtain an elastic Potts neural network that is loosely interconnected and has geometrical abstraction for the TSP problem. And this interactive mode method is surprisingly simple and efficient. The EPNN has advanatges in both of economical architecture and efficient computation for improving both of the original elastic ring method [2] and the Potts neural network [5]. We will omit lengthly reviews and directly present our network in the next section. Section III depicts simulation results of comparing the EPNN with the simulated annealing method. We will draw summary in the final section.

2   Elastic Potts Neural Network

The TSP problem can be described as, given N positions of cities in an Euclidean plane, what is the shortest path visiting each city once and returning to where it starts. The basic idea of the elastic ring method for the TSP problem is to design a non-uniform evolution of a close path to capture gometrical distribution of city set. This close path is gradually enlarged with the beads' neighbouring relationship preserved until the path eventually approaches sufficiently close to all cities and a tour is thus defined. Each bead on the close path follows the dynamics under the influence of two types of forces. The first is the attractive force which moves bead towards those cities which are nearest, the second is the restoration force which pulls the bead towards its neighbour beads on the path. This idea provides physical comprehension for treating the TSP problem.

In this work, we maintain the number of beads on the path fixed as the city number. The close path starts from a small circle. We first consider a neural network based on one to one mapping from beads to

cites under given distribution of the close path. The designed evolution of the close path is then following the bead to city mapping.

N Potts neurons are employed for N cities. Each Potts neuron contains N binary neural variables, denoted as $S_p = [S_{p1}, ..., S_{pN}]$. The neural variable S_{pi} is defined to be 1 when bead i is mapped to city p and 0 otherwise. The mapping is based on the following constraints: 1. One city p should occupy precisely one bead i. 2. Different cities p and q should not map to the same bead i. 3. Each bead should be located to exactly one city. 4. A bead should be enforced according to the city which is nearest to it and the bead's two closest neighbors. The first constraint can be formulated in the neural network in terms of Potts normalization conditions

$$\sum_i S_{pi} = 1 \tag{1}$$

for each city p. The other three constraints can be accomplished by energy penalty. By summing up each energy penalty with a Lagrange multiplier, the energy function specifying the bead to city mapping can be formulated as following.

$$
\begin{aligned}
E(\mathbf{S}) &= \frac{A}{2}\sum_p \sum_{i \neq j} S_{pi} S_{pj} + \frac{B}{2}\sum_i (\sum_p S_{pi} - 1)^2 + \sum_p \sum_i S_{pi}(d_{p,i-1} + 2d_{p,i} + d_{p,i+1}) \\
&= -\frac{A}{2}\sum_p \sum_i S_{pi}^2 + \frac{B}{2}\sum_i (\sum_p S_{pi})^2 + \sum_p \sum_i S_{pi} h_{pi} + \frac{1}{2}(A - B)N
\end{aligned}
\tag{2}
$$

where $d_{pi} = |x_p - y_i|$, $h_{pi} = d_{p,i-1} + 2d_{pi} + d_{p,i+1}$, and x_p and y_i denote the coordinates of city p and bead i, respectively.

In the next step, the mean field annealing theory is used to find the minimum of above energy function. Let the mean field variable V_{pi} represent $\langle S_{pi} \rangle$, the mean of S_{pi} as in [5]. The mean field equations for the energy function in equation (2) can be derived as follows.

$$
\begin{aligned}
V_{pi} &= F_N^i(U_p) = \frac{e^{U_{pi}}}{\sum_j e^{U_{pj}}} \\
U_{pi} &= -\frac{1}{T}\frac{\partial E(\mathbf{V})}{\partial V_{pi}} = \frac{1}{T}(AV_{pi} - B\sum_q V_{qi} + h_{pi})
\end{aligned}
\tag{3}
$$

Where T denotes the temperature which is gradually reduced and U_{pi} is the ith component of U_p. The mean field annealing theory implicitly transforms the discrete mapping to analog mapping. The mean field variable V_{pi} are analog values within zero and one. At each temperature , iterative execution of equation 3 brings the network to equilibrium state, denoted by $\mathbf{V^e} = \{V_{pi}^e\}$. The euqilibrium state represents an analog bead-city mapping with taking current external field $\{h_{pi}\}$ as criteria. The analog mapping is then used to move beads toward cities. The rule for the change Δy_i in the coordinate y_i of bead i is defined as follows.

$$\Delta y_i = \sum_p V_{pi}^e(x_p - y_i) \tag{4}$$

After changing the positions of beads, a new set of external fields $\{h_{pi}\}$ in equation (2) is obtained. The neural network then converges under new external fields at next reduced temperature. Above process operates under well thermal control, starting at a high temperature, and successively lowering it in the course of process. At sufficiently low temperature, $\mathbf{V^e}$ are hard limited to binary values that defines a tour and all beads are also close enough to all cities. The whole algorithm can be stepped as follows:

1. initializing T as T_0 and $y_i = \frac{1}{N}\sum_q x_q + r(\cos\frac{2\pi i}{N}, \sin\frac{2\pi i}{N})$, r is a random number generated between (0, 0.05).

2. setting $V_{pi} = \frac{\delta_{pi}+1}{N}$, δ_{pi} is a random number generated between (-0.01,0.01).

3. using equation 3 to obtain equilibrium state $\mathbf{V^e}$.

4. using $\mathbf{V^e}$ to update beads' positions as in equation 4.

5. setting T as T(1-α),if $\Sigma = \frac{1}{N}\sum_{pi}(V_{pi}^e)^2 < 0.5$, goto step 2, if $0.5 \leq \Sigma \leq 0.9$ goto step 3, otherwise end the process.

	mean	variance	total cpu for 200 runs
EPNN	7.796	0.0009	26200 sec
SA	7.874	0.02	1440 sec
SELF-ORG	7.864	0.0118	875 sec

Table 1: The statistical results of EPNN, SA and self-organization algorithm for 100-city TSP problem and the spent total CPU time on SPARC II workstation
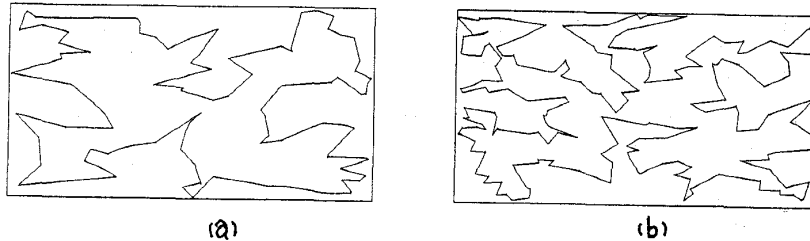

Figure 1: The solutions obtained by EPNN for 100-city (a) and 200-city (b) TSPs

3 Numerical Simulations

We find the above algorithm maintain highly stable quality. The average performance is better than that of simulated annealing[9] and the self-organization based algorithm[1]. And the best solution is also comparable to that of simulated annealing. These algorithm are implemented on SPARC II workstation. Table 1 shows the statistical results and the spent CPU time of these three algorithm for a 100-city TSP. The city set of the problem is randomly generated within a 1x1 square. The parameters of EPNN are set as $A = 0.5$, $B = 1$, and $\alpha = 0.01$. In our simulation for the SA, the tour generating procedure is a 2-opt procedure [7], temperature reducing condition is 1000 consequent tour rejections, and the algorithm terminating condition is 20000 consequent tour rejections. The thermal control for SA is the same as for EPNN. Within 200 executions, the obtained shortest tour lengths are 7.575 by SA, 7.776 by EPNN and 7.636 by self-organiztion algorithm. The average length by EPNN, 7.796, is shorter than SA's 7.874 and 7.864 of self-organization algorithm. In table 1, path length variance of EPNN is very small. This means high stability of the network behavior of the EPNN. Figure 1.a shows the best tour obtained by EPNN for the 100-city TSP.

The EPNN saves vast computations in comparison with the original Potts neural formulation [5]. Within 10 executions of original Potts Neural network, the best tour length is 8.189. The average tour legnth is 8.634. And the average CPU time for each execution is 27 min 30 sec, which is much greater than EPNN's 2 min 11 sec. Simulations of EPNN for larger size TSP run efficiently on workstation computer. Figure 1.b and figure 2 show the corresponding best solution of EPNN for 200-city TSP and 500-city TSP within 5 executions. The average CPU time for each execution is 14 min 20 sec for 200-city TSP and 2 hr 38 min 4 sec for 500-city TSP

4 Summary

An interactive mode of two physics-inspired model, elastic net[2] and Potts neural network[5], has been devised for solving TSP problem in this work. The developed EPNN is a natural abstraction of the elastic ring type method. Two important points of the EPNN are examined. At first, the EPNN possesses much loosely coupled interconnections. Each neural variable is only coupled with 2N-1 neural variables. This reduces half of interconnections in the original Potts neural formulation [5]. The second is that EPNN is the first Hopfield type neural network with geometrical abstraction. The energy term in all of other Hopfield type neural formulations for minimizing total distance is replaced by the dynamic external fields in EPNN. The external fields are encoded with two essential forces in elastic ring type method. These save vast numerical
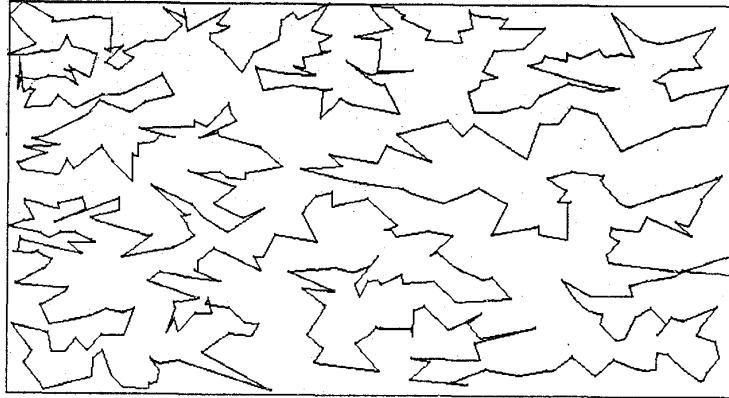
Figure 2: One solution obtained by the EPNN for 500-city TSP

computations in simulations and construct the design of neural network with geometrical abstraction for obtaining high quality solutions.

For the incorporation of elastic ring type method, the equilibrium state of the EPNN at each temperatue plays the role of relaxing the natural expansion for the close path. The evolution of the close ring path is a result of collective behavior of neural network. The bead number is fixed as the city number, which is much less than bead size in [1] [2]. And the adaption rule is very simple without the difficulty of parameter sensitivity to local minimum as in [2].

Generalizing current approach to other complex tasks is our most concern in future works.

References

[1] Angeniol B., Vaubois G. L. C. and Texier J. Y. L., : "Self-Organizing Feature Maps and the Travelling Salesman Problem", Neural Network, Vol. 1 (1988).

[2] Durbin R. and Willshaw,G., : "An analogue approach to the travelling salesman problem using an elastic net method", Nature, 326 (1987).

[3] Hopfield J.J. and Tank d. W., : "Neural computation of decisisions in optimization problems", Biol. Cybernetics, 52 (1985).

[4] Peterson C., : "Parallel distributed approaches to combinatorial optimization", Neural Comp., 2 (1990).

[5] Peterson C. and Söderberg B., : "A new method for mapping optimization problems onto neural network", Int. J. Neural Syst. 1, 3 (1989).

[6] Simic P.D., : "Statistical mechanics as the underlying theory of 'elastic' and 'neural' optimizations", Network 1 (1990).

[7] Gravey M.R. and Johnson D.S., : Computer and Intractability, A Guide to the Theory of NP Completeness, Freeman, San Francisco (1979).

[8] Yuille A. L., : "Generalized deformable model, statistical physics, and matching problems", Neural Comp. 2 (1990).

[9] Kirkpatrick S., Gelatt C.D. and Vecchi M.P., : "Optimization by simulated annealing", Science, 220 (1983).