The optimization for TSP using SOM method of many cities, for example 532 cities in USA

Kikuo FUJIMURA, Heizo TOKUTAKA, Shin-ichi TANAKA, Takatoshi MAENOU and Satoru KISHIDA

Tottori University
Department of Electrical and Electronic Engineering
4-101 Koyama, Tottori 680, JAPAN

Abstract

Kohonen's Self Organizing maps(SOM), were applied for the Traveling Salesman Problem(TSP). In the previous report, the simulations of 36 and 200 cities were carried out by SOM. From the report, the optimum moment values for both the cases were obtained and here, we found the relation between the values as $\beta = \gamma/M$(β: the momentum values, γ: the constant and M: the number of cities). Then, we have obtained the optimum moment value in the SOM for TSP of 532 cities of USA. The value also satisfies the relation of $\beta = \gamma/M$. By the relation, the optimum moment values can be estimated for more complicated $1,000$ and $10,000$ cities problems.

1 Introduction

The Traveling Salesman Problem (TSP) is one of the typical combination optimum problems, where all the prepared cities must be visited once with the shortest distance. The TSP was first challenged by Hopfield[1] with the interconnected network. The success made the neuro-computing method to become very famous. Angeniol et al[2] first applied the Kohonen's SOM (Self Organization feature Map)[3] to the TSP. Their algorithm is very simple and the application of their method to TSP (100, 200 cities) brought easily sub-optimum solutions with tolerant necessary time. We[4] improved the Angeniol's method[2] with several points, as one of which the renewal coefficient α decreased or increased with the increase or decrease of the total node number at the survey of the route of TSP respectively. The improvement[4] made the quality of the solution only slightly worse. However, the speed to obtain the solution became 5 times faster than the original method[2].
We modified the Angeniol's method considering the changes of the number of nodes during the survey of the solution, where the control parameter of the node is called as β. We applied the method to the problems of 36 and 200 cities. We found some relation of optimized β values between these 36 and 200 city problems. The relation is denoted as $\beta = \gamma/M$, where β:the momentum term, γ: the constant, and M: the number of cities. This relation is applied to the problem of the USA532 cities and the most optimized β was obtained. The value β satisfied the relation of $\beta = \gamma/M$. The relation could be applied to the more complicated problems of $1,000$ and $10,000$ cities problems to save the computing time.

2 Algorithm

2.1 The summary of the algorithm

The details of the algorithm are written in the original paper[2] and our improved method[4]. In the algorithm, all the nodes are connected on a one dimensional ring through the survey of TSP.

The nodes on the ring continuously move and are finally locked to the locations of the cities to obtain the most optimum or sub-optimum solutions. During the survey, all the nodes move freely on the plane where all the cities are fixed and located. When each city catches only one node on the ring, the survey will be finished and the actual path (solution) will be obtained. During the survey, each node J on the ring is renewed by the following eq.(1)

$$C_k^j \leftarrow C_k^j + f(G,n)(X_k^i - C_k^j) \tag{1}$$

where 2 dimensional positions of the i-th city and j-th node on the ring are expressed as (X_1^i, X_2^i). and (C_1^j, C_2^j), respectively. The nodes on the ring are also numbered from 1 to N (the total node number on the ring at the present search). The node J is counted clockwise or counter-clockwise from the node J_c which is the closest node to the i-th city in the present search and then the smaller number is taken as the order distance n. As shown in eq.(1), each node J becomes closer to the i-th city by the amount proportional to the renewal function $f(G,n)$ expressed as

$$f(G,n) = (\frac{1}{\sqrt{2}})exp(\frac{-n^2}{G^2}) \tag{2}$$

where G is the parameter adjusting the renewed neighbouring range. When G is large or small, the renewed neighbouring range becomes broad or narrow respectively. The value G in eq.(2) is determined appropriately at the start of the survey of TSP. After each search of all M cities during the survey, G will decrease following eq.(3)

$$G = \alpha G \tag{3}$$

We optimized the initial value G and the renewal coefficient α as 10 and 0.95 [4] respectively. Angeniol et al[2] fixed the value α during the survey. However, we[4] added the following momentum term to the renewal coefficient α as shown in eq.(4)

$$\alpha(t+1) = \alpha(t) + \beta[C(t) - C(t-1)] \tag{4}$$

where $C(t)$ is the total node number at the present search t of all M cities, $C(t-1)$ is the number at the previous search $(t-1)$, and β is an appropriate proportional coefficient. In our previous work[4], β is optimized as -0.001 for 200 cities. Here, β is negative and this means that the renewal coefficient α decreases (increases), when the total node number increases (decreases).

2.2 The further modification of the Algorithm

We already applied the algorithm which is described in the section 2.1, to the problems of 36 and 200 cities. The results of the 200 city problem are shown in Figure 1. In the figure, the average computing time by CPU Time, and the average Tour Lengths are obtained as a function of the values β. The CPU Time is normalized by the value of 30.62 sec in the case of $\beta = 0$. Tour Length is normalized by the shortest value of 11.0091 which is obtained in the case of $\beta = 0.001$. From the figure, the obtained Tour Length in the cases of $\beta = -0.001$ has an errors which is almost equal to an error of 5% in the case of $\beta = 0$. However, when the time is compared between $\beta = 0$ and $\beta = -0.001$, the case of $\beta = -0.001$ needs the 1/5 computing time which is shorter than the case of $\beta = 0$. The same results are obtained in the problem of 36 cities.

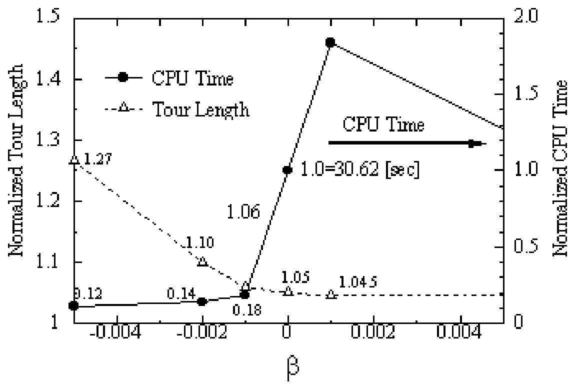

Figure 1: The trial experimental results of randomly located 200 city problem. The average Tour Lengths are normalized by the shortest Tour Length which is obtained in the experiment. The average CPU Times are also normalized by the average value at $\beta = 0$. The figure is enlarged at around $\beta = 0$.

Figure 2 shows the values of the momentum β which give the $1/5$ computing time compared with the case of $\beta = 0$, as a function of the number of cities. In the figure, both axes are drawn using the logarithmic scales. The results of 36 and 200 cities are elongated to further larger number of cities by a straight line. This straight line gives a relation of $\beta = \gamma/M$, where γ is a constant value. Then, the renewed eq.(4) is written as the following eq.(5).

$$\alpha(t+1) = \alpha(t) + \frac{\gamma}{M} \cdot [C(t) - C(t-1)] \tag{5}$$

From Figure 2, the value β can be read as -0.0004 at the city number of 532. Thus, a straight line of the relation of $\beta = \gamma/M$ in Figure 2 gives the values β of $1,000$ and also $10,000$ cities, very easily.

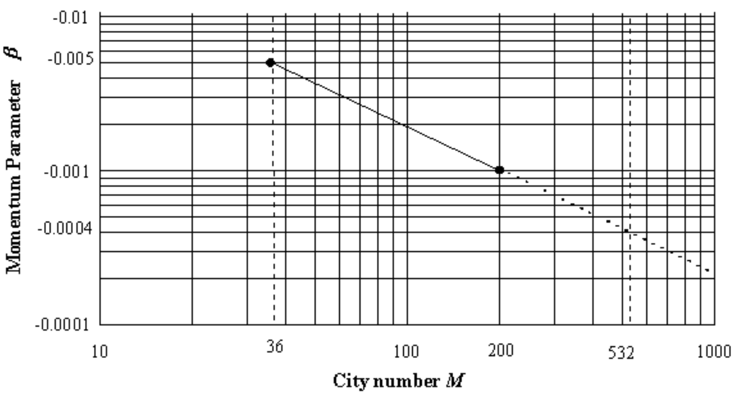

Figure 2: The relation between the momentum term β and the city number M.

3 The simulation and the results

The 532 city problem is reconfirmed as the benchmark simulation problem, because the city locations are always fixed and known, and the shortest traveling distance is already calculated by the super-computer[5]. One of our best results By the SOM method are shown in Figure 3.

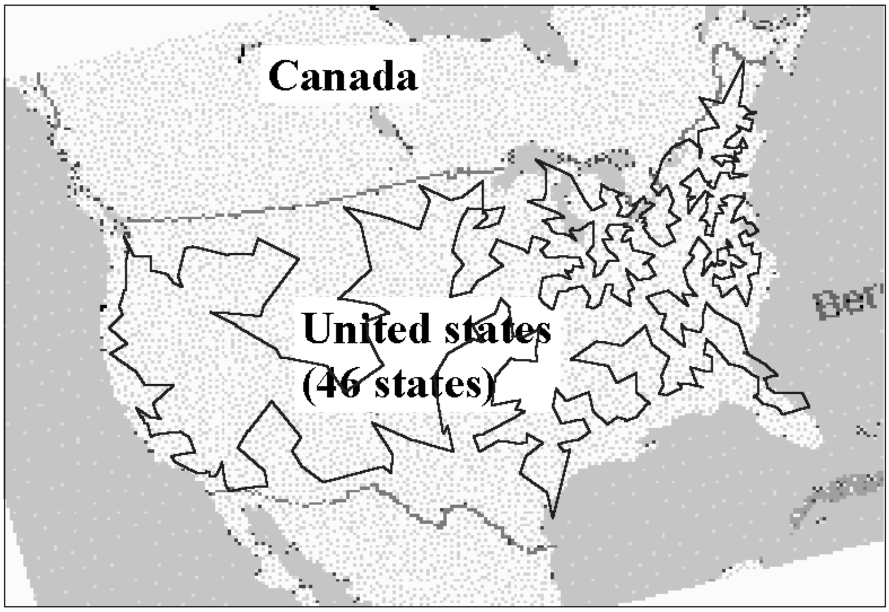

Figure 3: The most optimum Tour Length for the 532 city problem of USA obtained by our algorithm.

3.1 Experimental Conditions

In the experiment, we used a workstation (MicroSparcII, 85MHz), an initial node number $N = 1$ and the update coefficient $G = 1$, 5, 10 in eq.(3) are tried. We also tried $\beta = -0.1$, -0.01, -0.005, -0.002, -0.001, 0, 0.001, 0.01, 0.1. In the total 27 cases, each 100 tries were carried out, where the location of the initial node was determined by changing the seeds of the random numbers. However, in the Angeniol's method, the location of the initial node was always fixed. From our method, the shortest distance was obtained as $28,658$. However, the reference [5] gives $27,686$ as the theoretical shortest distance using the super-computer and quite longer CPU time. Our result is in the error of only 3%. Thus, we can obtain the second optimum solution very easily. The path of the tour is shown in Figure 3. When the values of N and G are fixed as 1 and 10, respectively, the Tour Length and the CPU Time were searched as a function of β. Figure 4 shows the results which has been enlarged at around $\beta = 0$. This is obtained by averaging 100 tries, whereby the initial node location is changed randomly for each try. Thus, the Tour Length and CPU time in Figure 4 show the averaged values of these 100 tries. The averaged Tour Length is normalized by the theoretical shortest distance $27,686$ and the averaged CPU Time is normalized at the value (22.0945 sec) of $\beta = 0$. From Figure 4, the computing time in the case of $\beta < 0$ is about $1/5$ compared with the case of $\beta = 0$. The results quite reasonably agrees with the cases of previous 36 and 200 cities. From the figure, we can understand that the averaged Tour Length can have the minimum value when β is in range of $-0.001 < \beta < 0.001$. When β becomes smaller over this range, the averaged Tour Length increases sharply. However, when β is positively large beyond the range, the distance slightly increases compared with the minimum value. The results also agree quite well with the results of 36 and 200 cities.

As the optimum solution with respect to the Tour Length and the CPU Time, $\beta = -0.0004$ is the recommended. The averaged Tour Length at $\beta = -0.0004$ is almost equal to the value at $\beta = 0$. The length is longer by only 5% compared with the theoretical value. However, the averaged CPU Time at the former case shows the $1/5$ computing time compared with the latter case.

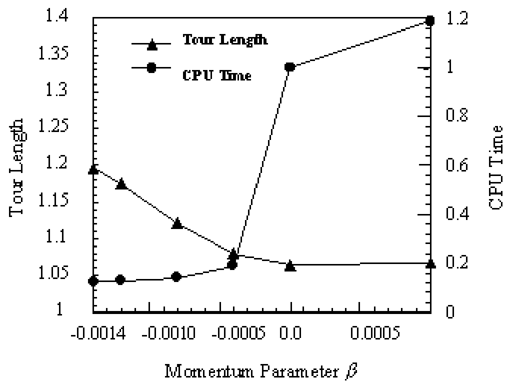

Figure 4: The experimental results of the 532 citiy problem of USA. The figure is enlarged at around $\beta = 0$.

3.2 Agreements between theoretical estimation and experimental results

From the results of 36 and 200 cities, it can be understood that the 1/5 computing time compared with the case of $\beta = 0$ can be obtained by the β value which comes from the relation of $\beta = \gamma/M$. From Figure 4 of 532 cities problem, the value $\beta = -0.0004$ causes the 1/5 computing time compared with $\beta = 0$. The value $\beta = -0.0004$ agrees quite well with the value predicted from Figure 2 at 532 cities position. Figure 5 is written and rescaled by the value γ of eq.(6) instead of the value β.

$$\gamma = \beta \times M \tag{6}$$

For 200 and 532 cities, $\gamma = 200\beta$ and $\gamma = 532\beta$, respectively. Using these γ values, Tour Length and CPU Time are written on the same figure of Figure 5. As predicted, both results of 200 and 532 cities agree quite well as shown in Figure 5. From Figure 5, it can be understood that $\gamma = -0.2$ gives 1/5 computing time compared with the case of $\gamma = \beta = 0$, and also with almost the same Tour Length.

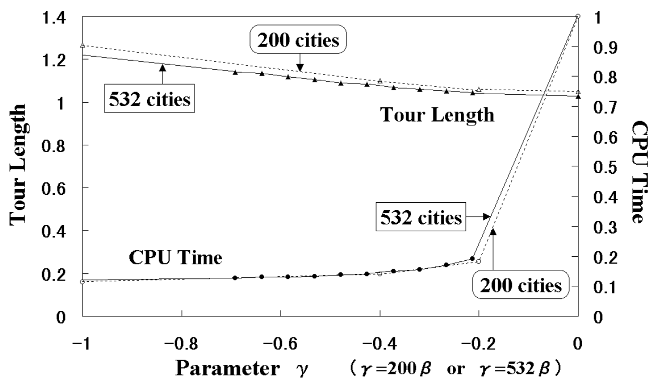

Figure 5: Comparison of the results of 200 and 532 city problems as a function of the new parameter γ.

$\gamma = -0.2$ will be effective for any city number to decide the momentum value β to survey the tour path. Using the relation of eq.(6), the most optimum momentum value β will be estimated for more complicated problems of 1,000 and 10,000 city number, which will be -2×10^{-4} and -2×10^{-5}, respectively.

4 Conclusions

As shown in this article, the Angeniol's method which uses the Kohonen's SOM method is very powerful in solving the Traveling Salesman Problem (TSP) of one of the typical combinatorial optimization problem, in shorter time compared with any other methods. In our previous study, we modified the Angeniol's method introducing the momentum term β to solve the TSP problem in more shorter computing time. The optimum values β were surveyed for 36 and 200 city problems. From the results of 36 and 200 cities, it can be understood that the value β which makes the 1/5 computing time compared with the case of the value $\beta = 0$ has the relation of $\beta = \gamma/M$, where γ becomes -0.2 and M is the number of cities. For the 532 cities problem of USA, the above relation of $\beta = \gamma/M$ is also confirmed. Using this defined value of β, the computing time became 1/5 compared with the case of $\beta = 0$. The Tour Length with this defined β value also became almost same compared with the case of $\beta = 0$. The average Tour Lengths of these cases are about only 5% worse compared with the true theoretical Tour Length [5]. Therefore, using the value β which satisfies the relation of $\beta = \gamma/M$, we can obtain the theoretically balanced optimum solution for the TSP problems of any number of cities.
This article shows that the relation of $\beta = \gamma/M$ can give the second optimum solution of TSP reducing the computing time very effectively. In fact, it will be possible to find the optimum momentum coefficients for $1,000$ and $10,000$ cities problems in order to reduce the computing time.

References

[1] J.J. Hopfield and D. Tank. Neural computation of Decisions in Optimization Problem. In *Biological Cybernetics*, volume 5, pages 141–152 1994.

[2] B. Angeniol, G. de La C. Vaubois and J.Y. Le Texier. Self-Organizing Feature Maps and the Travelling Salesman Problem. In *Neural Networks*, volume 1, pages 289–293 1988.

[3] T. Kohonen. *Self-Organizing Maps.* Springer-Verlag, 1995.

[4] K. Fujimura, H. Tokutaka, Y. Ohshima and S. Kishida. The Traveling Salesman Problem Applied to the Self-Organizing Feature Map. In *Proceedings of International Conference on Neural Information Processing (ICONIP'94-Seoul)*, pages 427–432 1994.

[5] M. Padberg and G. Rinaldi OPTIMIZATION OF A 532-CITY SYMMETRIC TRAVELING SALESMAN PROBLEM BY BRANCH AND CUT. In *Operation Research Lett.*, volume 6, pages 1–7 1987.