Chaotic Simulated Annealing with Augmented Lagrange for Solving Combinatorial Optimization Problems

Fuyu Tian

School of EEE
Nanyang Technological University
Block S2, Nanyang Avenue,
Singapore 639798
P144828100@ntu.edu.sg

Lipo Wang

School of EEE
Nanyang Technological University
Block S2, Nanyang Avenue,
Singapore 639798
elpwang@ntu.edu.sg

Abstract

Recent reports show that chaotic simulated annealing (CSA) can be successfully used to find the global optimum or near optimum with a set of parameters carefully chosen. However, CSA still uses a penalty term to enforce solution validity as in the Hopfield-Tank approach. This penalty method exhibits a conflict between solution quality and solution validity in the penalty approach. In addition, the relative magnitude of the penalty term often needs to be determined by trial-and-error. To overcome this disadvantage, we proposed a method which we call augmented Lagrange chaotic simulated annealing (AL-CSA). Simulation results on 48-city Traveling Salesman Problem (TSP) show that this method can maintain CSA's good solution quality while avoiding the need of penalty terms. Furthermore, convergence time is shorter compared to CSA. The influence of Lagrange multipliers on the process of searching for global minimum is also demonstrated.

1. Introduction

Recently, Chen and Aihara proposed a chaotic simulated annealing (CSA) method that can harness the advantage of both chaotic neurodynamics and conventional convergent neurodynamics [1-3]. They reported that the CSA has richer dynamics and higher ability of searching for globally optimal or near-optimal solutions. Their experimental studies demonstrate the effectiveness of the CSA and obtain good solutions for TSPs with a set of parameters chosen [1-3] much more easily compared to the Hopfield-Tank approach [4][5].

However, in CSA there still exists a difficulty in obtaining solutions of *good quality* and *valid* solutions simultaneously. In addition, one still needs to perform time-consuming, trial-and-error work to choose the relative sizes between the terms that represent solution quality and solution validity. This is because that CSA still belongs to the penalty method for constrained real optimization. In order to converge to a feasible solution with the penalty method, the weighting factor for the penalty term must be sufficiently large. However, as the penalty term becomes stronger, the role of the original objective function becomes relatively weaker. Solutions found in this way are affected more by the penalty terms and hence are less favorable in terms of the original objective. Furthermore, when the penalty term is too larger, the problem becomes ill-conditioned. Our previous experiments on traveling salesman problem (TSP) showed that CSA's solution quality is quite sensitive to the choice of penalty terms, especially when the number of cities becomes larger [20]. As the number of cities in the TSP becomes larger, it becomes more difficult for the network to find valid tours.

To completely avoid the ill-conditioned problems and effectively eliminate the unfavorable influence of the penalty terms on solution quality, we incorporate augmented Lagrange multipliers into the CSA, obtaining a method that we call augmented Lagrange chaotic simulated annealing (AL-CSA), following Li [11] who used Lagrange in the Hopfield-Tank approach to solve optimization problems. With the Lagrange multipliers, the constraints are satisfied exactly without the need of penalty terms. We apply the AL-CSA to solve 48-USA-city TSP [10]. In the AL-CSA, the influence of the penalty terms changes dynamically, the Lagrangian neurons lead the dynamic trajectory into the feasible solution region. Furthermore, our experiments show that the convergence time becomes shorter than that of CSA. The influence of the Lagrangian multipliers on the process of searching is also demonstrated in the end.

2. Augmented Lagrange-Chaotic Simulated Annealing (AL-CSA)

0-7803-6456-2/00/$10.00 ©2000 IEEE

A combinatorial optimization problem can be converted into the following constrained minimization:

minimize $E(x)$ (1)

subject to $C_k(x) = 0, \quad k = 1,...K$ (2)

where $x = (x_1, x_2, ..., x_n)^T \in R^n, E: R^n \to R$, $E(x)$ is the energy function, $C: R^n \to R^K$, $C = (C_1, C_2, ..., C_K)^T$ are real functions that represent some equality constraints and take the value of zero when the constraints are satisfied. E and C are assumed to be twice continuously differentiable. An augmented Lagrange function can be formed:

$$L_a(x, \lambda) = E(x) + \sum_k \lambda_k C_k(x) + \frac{1}{2}\sum_k a_k [C_k(x)]^2 \tag{3}$$

where $\lambda = (\lambda_1, \lambda_2, ..., \lambda_K)^T$ are Lagrange multipliers. $a_k > 0$ are finite weighting factors. The introduction of the quadratic term $\sum_k a_k [C_k(x)]^2$ with $C_k(x) = 0$ does not alter the location of the saddle point. In fact, this term can effectively stabilize the system.

The augmented Lagrange Chaotic Simulated Annealing (AL-CSA) model can be written as follows:

$$x_{ij}(t) = \frac{1}{1 + e^{-y_{ij}(t)/\varepsilon}} \tag{4}$$

$$y_{ij}(t+1) = k y_{ij}(t) - z(t)(x_{ij}(t) - I_0) - \alpha\left\{\frac{\partial E(x)}{\partial x_{ij}} + \sum_k \lambda_k \frac{\partial C_k(x)}{\partial x_{ij}} + \sum_k a_k C_k(x)\frac{\partial C_k(x)}{\partial x_{ij}}\right\} \tag{5}$$

$$z(t+1) = (1 - \beta) z(t) \tag{6}$$

$$\lambda_k(t+1) = \lambda_k(t) + a_k C_k(x_{ij}(t+1)) \tag{7}$$

In the above, $k, \alpha, \beta, \varepsilon$ are CSA's damping factor of nerve membrane $(0 \le k \le 1)$, positive scaling parameter for inputs, damping factor of the self-feedback connection weight $z(t)$ $(0 \le \beta \le 1)$ and steepness parameter of the output function, respectively.

3. Solving TSP Using AL-CSA

In Traveling Salesman Problem, the energy function is $E(x) = \frac{1}{2}\sum_i \sum_j \sum_{k \ne i} d_{ik} x_{ij}(x_{kj+1} + x_{kj-1})$, d_{ik} is the distance between city i and city k [4][5]. We use the following equality constraints that were used by Hopfield-Tank and other researchers:

$C_1(x) = \sum_i x_{ij} - 1 = 0, \forall j$, $C_2(x) = \sum_j x_{ij} - 1 = 0, \forall i$,

$C_3(x) = \sum_{l \ne j} x_{ij} x_{il} = 0, \forall i, j$, $C_4(x) = \sum_{k \ne i} x_{ij} x_{kj} = 0, \forall i, j$,

$C_5(x) = x_{ij}(1 - x_{ij}) = 0, \forall i, j$ (8)

Hence the total number of constraints is $n + n + n^2 + n^2 + n^2 = 3n^2 + 2n$. Here n is the number of cities in the TSP.

The three terms of (5) for the TSP are as follows:

$$\frac{\partial E(x)}{\partial x_{ij}} = \sum_{k \ne i} d_{ik}(x_{kj+1} + x_{kj-1}) \tag{9}$$

$$\sum_k \lambda_k \frac{\partial C_k(x)}{\partial x_{ij}} = \lambda_1 + \lambda_2 + \lambda_3 \sum_{l \ne j} x_{il} + \lambda_4 \sum_{k \ne i} x_{kj} + \lambda_5(1 - 2x_{ij}) \tag{10}$$

$$\sum_k a_k C_k(x)\frac{\partial C_k(x)}{\partial x_{ij}} = a_1(\sum_i x_{ij} - 1) + a_2(\sum_j x_{ij} - 1) + a_3 x_{ij}(\sum_{l \ne j} x_{il})^2 + a_4 x_{ij}(\sum_{k \ne i} x_{kj})^2 + a_5 x_{ij}(1 - x_{ij})(1 - 2x_{ij}) \tag{11}$$

We experiment on the 48-USA-city TSP that has been tackled by Chen and Aihara [3]. To compare the performance with the CSA, we use a set of $k, \alpha, \beta, \varepsilon$ that is the same as Chen and Aihara's [3]:

$k, \alpha, \beta, \varepsilon$ = 0.90, 0.015, 0.00005, 0.004; $I_0 = 0.50$; $z(0) = 0.10$.

The other parameters can be chosen as follows:

$\lambda_k(0) = 1$, $a_1 = a_2 = 0.0003$, $a_3 = a_4 = 0.00001$, $a_5 = 0$;

It should be noted that the above parameters can be chosen quite flexibly. Our simulations show that the AL-CSA is not sensitive to these parameters.

In the experiment, one iteration means that all neuron states are cyclically updated once. The neuron outputs are discrete as used by Chen and Aihara [3].

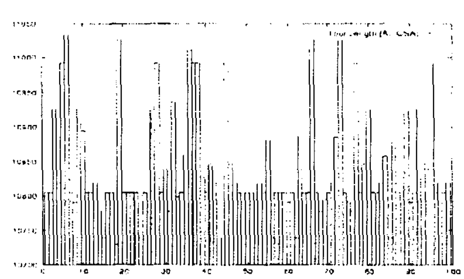

Figure 1. Tour lengths of 100 runs with different randomly selected initial network states using AL-CSA

Figure 1 shows the tour lengths of 100 runs with different randomly selected initial network state, by AL-CSA for the 48-USA-city TSP. The average tour length is 10839. The average iteration steps are 12016. As shown in Figure 1, AL-CSA can always converge to a feasible solution, i.e., with 0% invalid tour. In contrast, Chen and Aihara's CSA produces 5% invalid tours. It is therefore difficult to compare average tour length with their results. If the invalid tours are not included in averaging, their average tour length is 10805.7, which is comparable to ours. Their average time of iterations is 25632, which is much longer compared to ours. Hence the AL-CSA can converge much faster than CSA when solving large TSP's. The Lagrangian neurons lead the dynamic trajectory into feasible solution regions and contribute to the searching process. Certainly this additional Lagrangian neurons also make the network more complicated. The shortest tour length in our experiments is 10729 [Figure 2]. Due to the sensitivity to choice of parameters in CSA [20], we were not able to reproduce using CSA 5% global minimum (with tour length 10628) reported in [3].

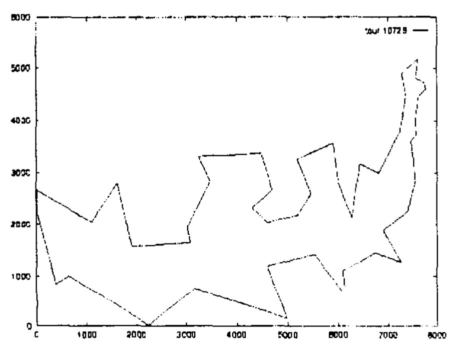

Figure 2. The shortest tour (10729) in our experiments

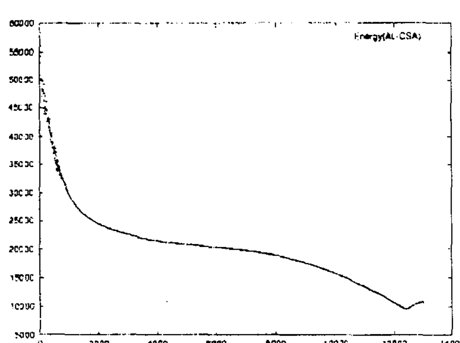

Figure 3. Lagrangian energy evolution process in AL-CSA

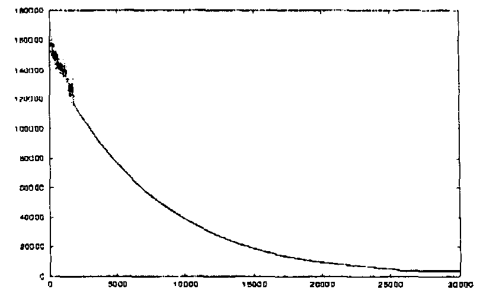

Figure 4. Lyapunov energy evolution process in CSA

Figure 3 shows the evolution of energy $L_a(x, \lambda)$ in equation (3). Its behavior is apparently different from the usual descent decreasing Lyapunov energy. Figure 4 shows the Lyapunov energy evolution process in CSA.

4. Summary

In this paper we proposed augmented Lagrange chaotic simulated annealing (AL-CSA), obtaining comparable good solution quality while eliminating the need to choose the penalty terms carefully. Our experiments also show that the convergence time becomes shorter than that of CSA when the city number of TSP is large. The solution validity can always be guaranteed, which is quite desirable for real-world applications. The influence of Lagrangian multipliers on the process of searching global minimum is also demonstrated in the end.

References

[1] L. Chen and K. Aihara, "Chaotic simulated annealing for combinatorial optimization," N. Aoki et al (eds.), *Dynamical Systems and Chaos*, Singapore, World Scientific, vol. 1, pp. 319-322, 1994.

[2] L. Chen and K. Aihara, "Transient chaotic neural networks and chaotic simulated annealing," M. Yamguti (ed.), *Towards the Harnessing of Chaos*. Amsterdam, Elsevier Science Publishers B.V. pp.347-352, 1994.

[3] 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.

[4] J. J.Hopfield and D. W. Tank, "Neural computation of decisions in optimization problems," *Biological Cybernetics*, vol. 52, pp. 141-152, 1985.

[5] G.V. Wilson and G. S. Pawley, "On the stability of the traveling salesman problem algorithm of Hopfield and Tank," *Biological Cybernetics*, vol. 58, pp. 63, 1988.

[6] K. Aihara, T. Takabe and M. Toyoda, "Chaotic neural networks," *Physics Letters A*, vol. 144, no. 6, 7, pp. 333-340, 1990.

[7] E. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines*, Chichester: John Wiley, 1989.

[8] D. P. Bertsekas, *"Constrained optimization and Lagrange multiplier methods,"* Academic Press, New York, 1982

[9] S. Zhang and A. G. Constantinides, "Lagrange programming neural networks," *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal processing*, vol. 39, no. 7, pp. 441-452, 1992.

[10] G. Reinelt, "TSPLIB-A traveling salesman library," ORSA Journal on Computing, vol. 3, pp. 376-384, 1991.

[11] S. Z. Li, "Improving convergence and solution quality of Hopfield-type neural networks with augmented Lagrange multipliers," *IEEE Transactions on Neural Networks*, vol. 7, no. 6, pp. 1-14, 1996.

[12] R. D. Brandt, Y. Yang, A. J. Laub and S. K. Mitra, "Alternative networks for solving the travelling salesman problem and the list-matching problem," in *IEEE International Conference on Neural Networks*, vol. 2, pp. 333-340, 1988.

[13] S. Abe, "Theories on the Hopfield neural networks," in *Proc. IEEE INNS International Joint Conference on Neural Networks*, vol. 1, pp. 557-564, 1989.

[14] L. Wang and D.L. Alkon, (eds.), *Artificial neural networks: Oscillations, chaos, and sequence processing*. (IEEE Computer Society Press), Los Alamitos, California, 1993.

[15] L. Wang, "Oscillatory and chaotic dynamics in neural networks under varying operating conditions," *IEEE Transactions on Neural Networks*, vol.7, no. 6, pp. 1382-1388, 1996.

[16] T. Kwok, K. Smith and L. Wang, "Solving combinatorial optimization problems by chaotic neural networks," C. dagli et al. (eds.) *Intelligent Engineering Systems through Artificial Neural Networks*, vol. 8, pp. 317-322, 1998.

[17] T. Kwok, K. Smith and L. Wang, "Incorporating chaos into the Hopfield neural network for combinatorial optimization," in *Proc. 1998 World Multiconference on Systemics, Cybernetics and Informatics*, N. Callaos, O. Omolayole, and L. Wang, (eds.) vol.1, pp. 646-651, 1998.

[18] L. Wang and K. Smith, "Chaos in the discretized analog Hopfield neural network and potential applications to optimization," *Proc. International Joint Conference on Neural Networks*, vol.2, pp. 1679-1684, 1998.

[19] L. Wang and K. Smith, "On chaotic simulated annealing," *IEEE Transactions on Neural Networks*, vol. 9, no. 4, pp. 716-718, 1998.

[20] F. Tian and L. Wang, "Solving optimization problems using transiently chaotic neural networks: choices of network parameters," *Proc. Second International Conference on Information, Communications & Signal Processing* , Singapore, Dec 7-10, 1999.