Hybrid Simulated Annealing Algorithm Based on Adaptive Cooling Schedule for TSP

Yi Liu
School of Computer Science and Technology, Wuhan University of Technology
csliuyi@163.com

Shengwu Xiong*
School of Computer Science and Technology, Wuhan University of Technology
xiongsw@whut.edu.cn

Hongbing Liu
School of Computer Science and Technology, Wuhan University of Technology
liuhbing@sohu.com

ABSTRACT

The traveling salesman problem(TSP) is one of the most notoriously intractable NP-complete optimization problems. Over the last 10 years, simulated annealing and tabu search have emerged as an effective algorithm for the TSP. However, the quality of solutions found by using tabu search approach depends on the initial solution and the iteration process of simulated annealing is slow. To overcome this problem and provide an efficient methodology for the TSP, the heuristic search approach based on simulated annealing which combining tabu search strategy and two neighborhood perturbation factor is developed. The proposed hybrid algorithm is tested on standard benchmark sets and compared with the conventional simulated annealing algorithm. The computational results show that the proposed algorithm has significantly better convergence speed compared with conventional simulated annealing algorithm and can obtain high-quality solutions within reasonable computing times.

Categories and Subject Descriptors

G.1.6 [**Numerical Analysis**]: Optimization—*Simulated annealing*

General Terms

Algorithms

Keywords

Simulated Annealing Algorithm, Tabu Search, Adaptive Cooling Schedule, Traveling Salesman Problem

1. INTRODUCTION

The traveling salesman problem (TSP)[1] is one of the most widely studied NP-hard combinatorial optimization problems. Its statement is deceptively simple, and yet it remains one of the most challenging problems in Operational Research. A lot of algorithms have been proposed to solve TSP [2][3][4]. Some of them (based on dynamic programming or branch and bound methods) provide global optimum solution. Other algorithms are heuristic ones, which are much faster, but they do not guarantee the optimal solutions. There are well known algorithms based on 2-opt or 3-opt change operators, Lin-Kerninghan algorithm (variable change) as well algorithms based on greedy principles (nearest neighbor, spanning tree, etc). The TSP was also approached by various modern heuristic methods, like simulated annealing, evolutionary algorithms and tabu search, even neural networks.

Tabu Search (TS)[5] is regarded as a higher-level heuristic approach used for solving optimization problems. The main feature of heuristic tabu-search is its ability to escape from the local optima encountered during the search by using the list of prohibited neighboring solutions known as tabu-list (Swarnkar and Tiwari, 2004).The main drawback associated with tabu search is cycling, in which it follows the same path unless a tabu neighbor exists.Beside that,TS's solution quality is highly dependent on initial solution.

Simulated Annealing(SA)[6] is a generic probabilistic meta-algorithm for the global optimization problem, namely locating a good approximation to the global minimum of a given function in a large search space. By contrast to TS, SA which is analogous to the physical process of annealing possesses a formal proof of convergence. The behavior of SA can be controlled by the cooling schedule and is not dependent on the initial solution. For certain problems, SA may be more effective than exhaustive enumeration - provided that the goal is merely to find an acceptable good solution in a reasonable amount of time, rather than the best possible solution. The SA algorithm with all its advantage also has some demerits, such as it requires large number of iterations to generate an optimal or near optimal solution, and it's initial temperature is difficult to determine.

On the basis of the existing SA algorithm, the paper proposed a hybrid tabu-simulated approach which introduces a new cooling schedule to solve TSP. With this new temperature control scheme, the search is given a higher chance of an uphill move at the end of search. Combined with the excellent local search ability of TS and the fine solution quality of SA. The proposed approach is less dependent on the specific information of problem compared with conventional SA.

*The corresponding author.

GEC'09, June 12–14, 2009, Shanghai, China.
ACM 978-1-60558-326-6/09/06.

2. TABU-SIMULATED ANNEALING

It has been seen that tabu search algorithm uses short term memory of recently visited solution known as tabu list to escape from local optima, but tabu list has a deterministic nature and thus cannot avoid cycling. This drawback of tabu search has been taken care by SA in which stochastic characteristic avoids cycling. Because SA has no memory of recently visited solutions, the rate of improvement of solution is very slow. There is always a probability for the search to return to the same solution again. However, with the help of a short-term memory, the search can be restricted from retiring to a previously visited solution and performance of SA can be enhanced significantly. Keeping in mind the above ideas, a hybrid algorithm called tabu-SA has been proposed in this paper. This algorithm is a SA approach supplementary with a tabu list. Such an algorithm for TSP problem is given in more detail in the following section.

2.1 Cooling Schedule

In this section, we propose a modified version of the simulated annealing algorithm for TSP problems and we call it adaptive SA. The distinct feature of this method is the temperature update mechanism, which is an important part of the transition probability equation. In conventional simulated annealing, the temperature declines according to the following function:

$$T_{k+1} = \alpha T_k \tag{1}$$

Where T_k is the temperature in iteration k, α is a coefficient that controls the rate of temperature decline. The search begins with a high temperature allowing a higher chance of transition to a worse solution. By doing so, the search is able to move out of local minima. However, as the search continues, the temperature continuously declines resulting in a reduced chance of uphill transition. Such an approach could be useful if the local minima are near the start point, but may not lead to a near optimal solution if some local minima are encountered at a relatively low temperature toward the end of the search.

To alleviate this difficulty, we propose an adaptive simulated annealing method that takes into consideration the characteristics of the search trajectory. In this method instead of the previous cooling schedule (Eq. (1)) that is monotonically non-increasing, we use an adaptive cooling schedule that adjusts the temperature dynamically based on the profile of the search path. Such adjustments could be in any direction including the possibility of reheating. In our proposed method, the temperature is controlled by a single function that maintains the temperature above a minimum level. The heating process gradually takes place if there is any upward move, but the cooling is sudden with the first downhill move. In the proposed method, the following temperature control function is used.

$$\theta_i = \theta_{min} + \lambda \ln(1 + r_i) \tag{2}$$

where θ_{min} is the minimum value that the temperature can take, λ is a coefficient that controls the rate of temperature rise, and r_i is the number of consecutive upward moves at iteration i. The initial value of r_i is zero, thus the initial temperature $\theta_0 Comparison of convergence speed = \theta_0$. The purpose of the minimum temperature θ_{min} is twofold. First it prevents the probability function (Eq. (2)) from becoming invalid when r_i is zero. Second, it determines the initial value of the temperature. θ_{min} can take any value greater than zero. The rate of temperature rise is controlled by parameter λ.The value of λ corresponding to the temperature rising rate. When a large value is assigned to λ, the search spends less time looking for good solutions in its current neighborhood. Similarly, by assigning a small value to the parameter λ, the search spends more time looking for better solutions in the neighborhood. Tuning a value for the parameter λ could be linked to computation time, which also depends on the size and complexity of the problem[7].

At any point, if a move is made to a solution with a cost higher than that of the previous solution during the search process, i.e., upward move, then the counter r_i increases by 1. If the new solution has an equal cost, r_i remains unchanged; and if the cost is improved, then r_i is equal to zero.

In conventional SA, such a shortcoming could be alleviated if the proper search parameters, such as initial temperature, cooling rate and the length of Markov chain, are selected through a pre-search analysis. While, for the adaptive tabu-SA, it remains an advantage that less pre-search analysis may be required. In this study, we have developed an algorithm based on the above cooling schedule. From now on, we refer to this algorithm as tabu-simulated annealing or simply TS-SA algorithm.

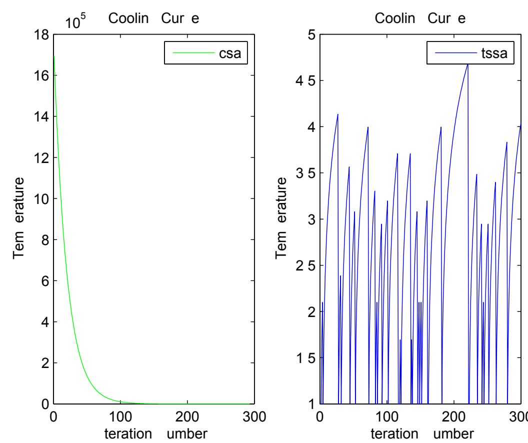

Figure 1: Cooling cure of SA and TS-SA

2.2 Neighborhood Generation

A neighborhood structure is a mechanism which can obtain a new set of neighbor solutions by applying a small perturbation to a given solution. Each neighbor solution is obtained immediately from a given solution by a move. Neighborhood structures and move evaluation strategies are directly effective on the efficiency of the local search for the TSP. Currently, the most well-known neighborhood generation method for the TSP problem is S. Lin' s 2-interchange method[8]. And the experimental evaluations show that the sampling method of reversing and moving a sub-permuta tion [9] has a highest efficiency for solving and is superior to the well-known S. Lin' s 2-interchange method, hence in the proposed algorithm, we use this perturbation method for the neighborhood generation, and we denoted this perturbation method as M_6. If X_i denotes the current path of

TSP problem, and $X_j = N(X_i)$ denotes the neighbor solution generated by a small perturbation, the M_6 perturbation method is:

$$X_i : 1 - \underline{2 - 3 - 4 - 5} - 6 - 7 - \underline{8} - 9 - 10$$

$$X_j : 1 - 6 - 7 - 8 - \underline{5 - 4 - 3 - 2} - 9 - 10$$

The basic perturbation method of exchanging two consecutive city's position is also applied in TS-SA, we named this method M_1, that is:

$$X_i : 1 - 2 - 3 - 4 - \underline{5 - 6} - 7 - 8 - 9 - 10$$

$$X_j : 1 - 2 - 3 - 4 - \underline{6 - 5} - 7 - 8 - 9 - 10$$

While the M_6 neighborhood perturbation method has extensive search ability and always has a good chance to direct the searching process into new area, its more subversive perturbation methodology in path sequence is disadvantageous to the exploitation and intensification search of near optimal path region. Compared with M_6 method, the M_1 neighborhood perturbation method's relative little impact on sub-path of near optimal path is favorable to the local search stage of the whole searching procedure. Considering the fact that most of near optimal path in TSP problem has similar sub-path in their path sequence, we employed the M_6 method as new path generation operator at the beginning of the search process and when the searching process falls into near optimal path area, M_1 method has taken effect to intensify the local search process aimed to find the global optimal path.

Algorithm1. TS-SA algorithm

Step1: Generating initial path X_0, initial temperature $T_0 = 1.0$, initializing TabuList $TL = NULL$, $k = 0.75$.

Step2: If running time k*total running time, generating new neighboring solution $X_j = N(X_i)$ by M_6 method, else generating new neighbor solution $X_j = N(X_i)$ by M_1. Compute $\Delta C = D(X_j) - D(X_i)$

Step3: Update R_i and TL, compute new temperature T_i

Step4: If $C < 0$, go to Step 6,else go to Step5

Step5: If $Exp(-\Delta C/T_i) > Random(0, 1)$, then go to Step 6, Else go to Setp8

Step6: If $X_i \in TL$, then go to Step8, else go to Step7

Step7: Accept the new path: $X_i = X_j$.

Step8: If running time satisfied the requiring time, then STOP, else go to Step2.

2.3 Tabu-list

The basic role of tabu list is to avoid the search process turning back to the solutions visited in the previous steps and the tabu tenure is the length of this list[10]. At each step of algorithm, tabu list checks whether the solution generated is latterly visited or not and thus help in restricting the algorithm to revisit the pre-visited solutions. This feature of TS-SA algorithm reduces the computational time of algorithm to reach the global optima. According to Knox's research [11], when the city count n not exceed 100, the length of tabu tenure has linear relationship with the city count n.

The steps of the proposed algorithm are described as Algorithm1.

Table 1: Numerical result of TSPLib

Problem	RT	opt.	meth.	best	Mean	PAB
bays29	5	2020	csa	2020	2057.8	1.8%
			tssa	2028	2035.8	0.8%
gr48	10	5058	csa	5125	5295.5	4.7%
			tssa	5177	5235.0	3.5%
eil76	10	538	csa	557	573.0	6.5%
			tssa	542	564.0	4.8%
berlin52	10	7542	csa	7658	7809.2	3.5%
			tssa	7648	7718.5	2.3%
eil51	10	426	csa	426	445.3	4.5%
			tssa	430	432.5	1.5%

3. NUMERICAL EXPERIMENTS

We employ the conventional SA algorithm as a benchmark for the performance evaluation of the proposed method and adopt the standard TSPLib's test instances bays29, gr48, eil51, berlin52, eil76 as test problems(available at http://www iwr.uniheidelberg.de/groups/comopt/software/TSPLIB95/tsp/) and choose the sequence path $1-2-3-4-5-6-\ldots-n$ as initial path for each of algorithms. The mean value of path and PAB (percentage above the best published solution) of each algorithm are chosen as indexes of performance for comparison.

$$\text{PAB} = \frac{\text{currentsolution} - \text{bestpublishedsolution}}{\text{bestpublishedsolution}} \times 100\%$$

The settings of substantial parameters of the conventional SA, TS-SA algorithm are as follow:

Neighborhood perturbation method adopts the M_6 method for the 75% of running time, and the M_1 method for the rest time of search.

Conventional SA's cooling schedule is updated by $T_{k+1} = 0.95 * T_k$, in which T_{k+1} represents the reduced new temperature, T_k represents the temperature before.

Initial Temperature is set to 1 for the TS-SA algorithm, tabu tenure is set to the same as in literature [12] $n \times 2.375$, n stands for the city count.

The RT in Table1 stands for the running times of two algorithms,and the Mean stands for the mean value of the calculation result.According to the Table 1, TS-SA algorithm has better average solution quality and lower PAB when it comes to the five benchmark instances of TSPLib.

It can be seen from the Figure 2 that TS-SA algorithm has a faster convergence rate compared with conventional SA algorithm because of the adoption of new cooling schedule. This is more evident when it comes to larger problem such as eil51. Figure 2 illustrated that at the end stage of searching, TS-SA algorithm has a better chance of upward move that causes the convergence curve progress more turbulent than conventional SA algorithm so that it can direct itself to more extensive searching area to avoid the freeze

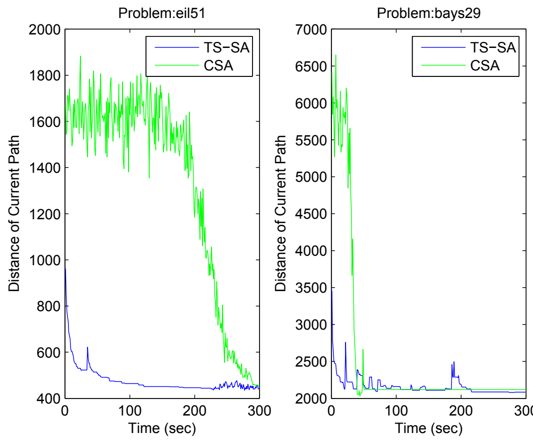

Figure 2: Comparison of convergence speed

state of conventional SA algorithm. Due to the fact that we utilized the M1 neighborhood perturbation method, Figure 2 also shows that the upward move in the late stage of TS-SA algorithm has been controlled within an appropriate range to guarantee that the searching process still falls into the near optimal path area.

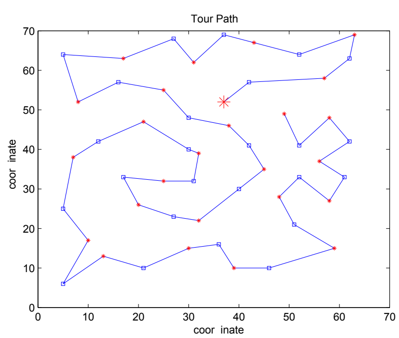

Figure 3: Obtained optimal tour path of eil51 problem by TS-SA, total distance:426

4. CONCLUSIONS

In this study, we proposed two adaptations to the standard simulated annealing based on a new temperature control scheme in conjunction with a tabu list to solve the TSP problem. With this new temperature control scheme coupled with two neighborhood generation factor, the search is given a higher chance of an uphill move at the end of search in order to escape from local minima. The performance of proposed TS-SA algorithm on test problem reported in literature is found to be superior to conventional SA algorithm with respect to convergence speed and average solution quality. By applying a tabu list, the TS-SA algorithm has conspicuously faster convergence rate compared with conventional SA. With the strategy of using different neighborhood generation factor at different stage of the search, the search has a higer probability to jump out of the big valley in search trajectory contrast with conventional SA.

Despite the advantages mentioned above, there is still chance to explore worse solutions region at the end of search for TS-SA algorithm. These facts are issues that still needs careful and further consideration.

5. ACKNOWLEDGMENTS

This work was supported in part by National Science Foundation of China (Grant No.40701153)and Wuhan International Cooperation and Communication Project (Grant No.200770834318).

6. REFERENCES

[1] R. Durbin and D. Willshaw. An anlaogue approach to the traveling salesman problem using an elastic net approach. *Nature*, 326(6114):689–691, 1987.

[2] T.Guo and Z.Michalewize. Inver-over operator for the tsp. In *Proceedings of the 5th International Conference on Parallel Problem Solving from Nature*, pages 802–812. IEEE, July 1998.

[3] Z.C.Huang, X.L.Hu, and S.D.Chen. Dynamic traveling salesman problem based on evolutionary computation. In *Proceedings of Congress on Evolutionary Computation*, pages 1283–1288. IEEE, May 2001.

[4] Aimin Zhou, Lishan Kang, and Zhenyu Yan. Solving dynamic tsp with evolutionary approach in real time. In *Proceedings of Congress on Evolutionary Computation*, pages 951–957. IEEE, December 2003.

[5] F. Glover. Future paths for integer programming and links to artificial intelligence. *Computers and Operations Research*, 13(5):533–549, 1986.

[6] C. D. Gelatt Jr S. Kirkpatrick and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[7] Nader Azizi and Saeed Zolfaghari. Adaptive temperature control for simulated annealng:a comparative study. *Computers and Operations Research*, 13(31):2439–2451, 2004.

[8] S. Lin and B.W. Kernighan. An effective heuristic algorithm for the traveling-salesman problem. *Computers and Operations Research*, 21(2):498–516, 1973.

[9] P. Tian, H. Wang, and D. Zhang. Solving the travelling salesman problem by simulated annealing. *Journal of Shanghai Jiaotong University*, 29(12):111–116, 1995.

[10] F. Glover and M. Laguna. *Tabu Search*. Kluwer Academic Publishers, Dordrecht, 1997.

[11] J. Knox. Tabu search performance on the symmetric tsp. *Computers and Operations Research*, 21(8):786–802, 1994.

[12] T. H. Wu and C. C. Chang. A tabu search heuristic for the traveling salesman problem. *Journal of Da-Yeh University*, 19(1):87–99, 1997.