Global Analysis and Discrete Mathematics
Volume 2, Issue 1, pp. 33–38
ISSN: 2476-5341

Markov Chain Anticipation for the Online Traveling Salesman Problem by Simulated Annealing Algorithm

Gholam Hassan Shirdel · Mohsen Abdolhosseinzadeh

Received: 17 February 2016 / Accepted: 18 November 2016

Abstract The arc costs are assumed to be online parameters of the network and decisions should be made while the costs of arcs are not known. The policies determine the permitted nodes and arcs to traverse and they are generally defined according to the departure nodes of the current policy nodes. In on-line created tours arc costs are not available for decision makers. The on-line traversed nodes are fixed and unchangeable for the next times. A discrete time Markov chain is established in on-line policy times. Then, the best state is selected to traverse the next node by a simulated annealing heuristic.

Keywords Online travelling salesman problem · Online decision problem · Discrete time Markov chain · Simulated annealing

Mathematics Subject Classification (2010) 90B15 · 68W25

1 Introduction

In on-line stochastic network optimization problems decisions are made sequentially over partial information. On-line made decisions are not changed in the future; however, they should be feasible for the problem. The symmetric travelling salesman problem (S-TSP) is NP-hard [4]. However, there are some approximation algorithms where the arc costs satisfy the triangular inequality

G. H. Shirdel
Department of Mathematics, Faculty of Basic Science, University of Qom, Qom, Iran
Tel.: +98-25-32103022
Fax: +98-25-32854973
E-mail: shirdel81math@gmail.com

M. Abdolhosseinzadeh
Department of Mathematics, Faculty of Basic Science, University of Qom, Qom, Iran

[3,4]. In our considered on-line model the exact costs are not revealed until the end of the optimization process; we use the expected values as the given advanced information. Let $G = (N, A)$ be a complete undirected network with node set N and arc set A. For any $(i, j) \in A$, the cost from i toward j is equal to the cost of reverse direction, $c_{ij} = c_{ji}$, also $c_{ij} \leq c_{ik} + c_{kj}$ for any $(i, j) \in A$ and $k \in N$. In proposed on-line model the costs are not known; however, the expected values of the costs are given by arrival a node.

Initially, an approximated solution is created by Christofides algorithm [3]. In on-line manner, it must be decided to move from the current node toward the next node starting from the given source node and to return to it finally after traversing all the nodes exactly once. The on-line routing policy determines the order of nodes, and the made decisions could not be changed or rejected. Ausiello et al. [2] and Wen et al. [11] assumed to traverse a number of nodes, also Jaillet and Lu [6] considered some penalties for not served ones. Ausiello et al. [1] and Zhang et al. [13] supposed that it is possible to visit a node more than once. In our proposed model, it is restricted to traverse any node exactly once and to return to the source node.

We establish a discrete time Markov chain (DTMC) according to the on-line policies and the uniformly distributed transition probabilities. The states of DTMC are feasible tours. Then, the simulated annealing (SA) is applied to obtain a good improvement of the initial approximated solution. Initially a ρ-approximated solution is applied. Christofides [3] produced a 3/2-approximation solution, so it is the initial state of the established DTMC.

2 The established discrete time Markov chain

A DTMC is established at any on-line policy time that should be decided the next traversed node in the on-line tour. So, in any transition from the current state toward a new state exactly one arc is traversed. The process is ended when all the nodes are traversed exactly once and it is returned to the source node. The state $S_{t,k}$, for $t = 1, ..., n-2$ and $k = 1, 2, ..., n-(t+1)$, is a set of nodes those are created a tour. The initial state contains the pre-obtained approximated solution and its first node is fixed (the source node). A fixed node is defined as the node which was traversed previously, at the current on-line policy time t. Only one node could be fixed at any time. The next state $S_{t+1,j}$ is created just by single permutation of the unfixed nodes of the current state $S^*_{t,i}$. So, if v_{k-1} is the last fixed node of the state $S^*_{t,i} = \{1 = \bar{v}_1, \bar{v}_2, \ldots, \bar{v}_{k-1}, v_k, v_{k+1}, \ldots, v_n, 1\}$, then the state $S_{t+1,j}$ is created by permutation of node v_k with unfixed nodes $v_{k+1}, v_{k+2}, \ldots, v_n$ ($|N| = n$).

Theorem 1 *There is no repeated state in the created states of the on-line established DTMC for $n \geq 7$.*

Proof. The initial state $S_{0,1}$ contains the pre-obtained approximated solution and its first node is fixed. There is not any repeated state among those are created by an unfixed index. The source node is the first and the last node of the created states; so, the permutation is circular and any state could be accounted in its opposite direction. Whereas, there are at least two fixed indices, the permutation does not cause a repeated state because it is not possible double/more permutation. A repeated state does not occur after index $2n-5$ (it is double permutation), however, it may be occurred before.

The initial state could be repeated in index $n-2$, so $v_{n-1} = v_3$ and $\{1 = \bar{v}_1, v_2, v_3, \ldots, v_{n-1}, v_n, 1\} \equiv \{1 = \bar{v}_1, v_n, v_3, \ldots, v_{n-1}, v_2, 1\}$, then $n-1 = 3$ and $n = 4$. The state of index $n-2$ (where v_{k_2} is unfixed) could be repeated through the states from $n-1$ to $2n-5$ (where $\bar{v}_{k_2}$ is fixed). Suppose index r is a repeated state, $\{1 = \bar{v}_1, v_n, v_3, \ldots, v_{n-1}, v_2, 1\} \equiv \{1 = \bar{v}_{l_1}, \bar{v}_{l_2}, v_{l_3}, \ldots, v_{l_{n-2}}, v_{l_{n-1}}, v_{l_n}, 1\}$; so, $l_1 = \bar{v}_1 = 1, l_{n-1} = v_3$ and $v_{l_n} = v_2$, if $v_4 = \varnothing$ then $l_{n-2} = \varnothing$ and $n = 5$, otherwise if $v_4 \neq \varnothing$ then $l_{n-2} = v_4$ and $n = 6$. □

The transitions are possible from the previous on-line decided state ($S^*_{t,i}$) toward the created states of the current on-line policy time ($S_{t+1,j}$) by an unfixed node permutation. $p_{i,j}$ is the transition probability from $S^*_{t,i}$ toward $S_{t+1,j}$ and $p_{i,j} = p_{j,i}$. Let $I_{S^*_{t,i}}$ be the index set of the created states by $S^*_{t,i}$, then the transition probabilities are defined uniformly $p_{i,j} = 2/(n-(t+1))(n-t)$, for $j \in I_{S^*_{t,i}}$, and otherwise $p_{i,j} = 0$. Polychronopoulos and Tsitsiklis [10] proposed a Markov chain that its states show pairs of location and information. We consider directly the on-line traversed nodes, and the expected improvement determines the next node to traverse.

3 The Simulated annealing heuristic

Although it has been shown Markov decision problems could be solved in polynomial time [8], computations grow exponentially in practice [9]. So, we apply a Markov chain form of the SA over the created states by DTMC [9].

Suppose $\bar{C}_{S_{t,j}}(T)$ is the expected cost of the state $S_{t,j}$ when the temperature is T, then $\triangle\bar{C}_{i,j}(T) = \bar{C}_{S_{t+1,j}}(T) - \bar{C}_{S^*_{t,i}}(T)$ is the expected cost difference of the transition from $S^*_{t,i}$ toward $S_{t+1,j}$. The acceptance probability $A_{i,j}(T)$ is the probability that $S_{t+1,j}$ is accepted, when the on-line policy determined to be in $S^*_{t,i}$ previously. By Metropolis rule, the acceptance probabilities are defined $A_{i,j}(T) = exp(-\triangle\bar{C}_{i,j}(T)/T)$, for $\triangle\bar{C}_{i,j}(T) > 0$, and otherwise $A_{i,j}(T) = 1$. In the case $\triangle\bar{C}_{i,j}(T) > 0$, the state $S_{t+1,j}$ is accepted if by producing a random number λ, then $\lambda \leq exp(-\triangle\bar{C}_{i,j}(T)/T)$. For the SA the transition probability $M_{i,j}(T)$ form $S^*_{t,i}$ toward $S_{t+1,j}$ is obtained as $M_{i,j}(T) = p_{i,j}A_{i,j}(T)$, for $i \neq j$, and otherwise $M_{i,j}(T) = 1 - \sum_{k\neq i} p_{i,k}A_{k,j}(T)$. If it is decided currently to be in $S^*_{t,i}$, then $q_{i,j}(T)$ is the static probability that the on-line policy decides to be in the state $S_{t+1,j}$. The static probabilities are the limiting-state probabilities in the steady state analysis [5].

The initial temperature is set to be enough high, such that all of the created states are accepted those could improve the initial approximated solution. Afterwards, the temperature is determined practically to traverse all the nodes and to return to the source node. To determine the annealing parameters the proposed method by Park and Kim [7] is applied, however we change them to reach exactly $n-2$ iterations. The temperature is changed if there is no state that $\triangle \bar{C}_{i,j}(T) \leq 0$, then it is decreased with the rate $\alpha = T_0^{-1/(n-3)}$, where T_0 is the initial temperature and so $T := \alpha T$.

4 The Competitive analysis of the on-line made decisions

To minimize the expected cost of the on-line tour, the decision is done according to the expected costs of the current policy nodes. It is not allowed to traverse toward the states, which their expected costs exceed the initial ρ-approximation solution cost ($\rho > 1$). Now, let C^* be the optimal off-line cost, C^O be the optimal on-line cost and $\bar{C}_0$ be the initial expected cost, then $C^* \leq C^O \leq \bar{C}_0 \leq \rho C^*$. For two successive costs $C^*_{S_{t,i}}(T)$ and $C_{S_{t+1,j}}(T)$ such that $\triangle C_{i,j}(T) > 0$, we have $C^* \leq C_{S^*_{t,i}}(T) \leq \bar{C}_0 \leq \rho C^*$, and $C^* \leq C_{S_{t+1,j}}(T) \leq \bar{C}_0 \leq \rho C^*$, then $C_{S_{t+1,j}}(T) - C_{S^*_{t,i}}(T) \leq \bar{C}_0 - \bar{C}^* \leq \bar{C}_0 - \frac{1}{\rho}\bar{C}_0$. In other words, $exp(-\triangle C_{i,j}(T)/T) = exp(-(C_{S_{t+1,j}}(T) - C_{S^*_{t,i}}(T))/T) \geq exp(-(\bar{C}_0 - C^*)/T) \geq exp(-((\rho-1)/\rho)\bar{C}_0)$.

Ausiello et al. [2] gave a 2-competitive algorithm. Jaillet and Lu [6] used advanced information to present a 2-competitive algorithm on the real line, 2.28-competitive for general metric space and also a $(1.5\rho + 1)$-competitive where ρ is the approximation ratio; also, Yu et al. [12] presented a $(1+\rho)$-competitive algorithm. Ausiello et al. [1] gave a $((3+\sqrt{5})/2)$-competitive algorithm as the best possible one. Zhang et al. [13] provided a $(1+k)$-competitive non-polynomial and another $(4+k)$-competitive polynomial lower bounds that k is the number of blockage arcs. Wen et al. [11] gave $2L+1$-competitive ratio in the case of line segment $[-L, L]$ and 2-competitive where every arc has unit weight.

5 Experimental results

All results are obtained by MATLAB 7.6 in Dell Latitude E5500, with Intel Core 2 Duo CPU 2.53 GHz and 1.95 GB of RAM. We use the instance networks with Euclidean distances presented by TSPLIB available at [14]. The arc costs are computed with 2D Euclidean distance function and the nearest integer function is applied on the obtained solutions. For brd14051, d2103, d15112, d18512, fl1577, fl3795, rl5915, rl5934, rl11849 and usa13503 instances those an interval is given instead of the exact optimal value, we use the lower bound instead of the optimal offline value. The given distances are considered as the expected values according to the uniform distribution of the arc cost. The overall average results show the initial competitive ratio is 1.3937 and the obtained competitive and reverse competitive (of the reverse tour) ratios are 1.3087 and 1.2994, respectively.

The detailed results of the competitive ratios for the instance networks obtained by the initial approximated solution are shown in Figure 1. The

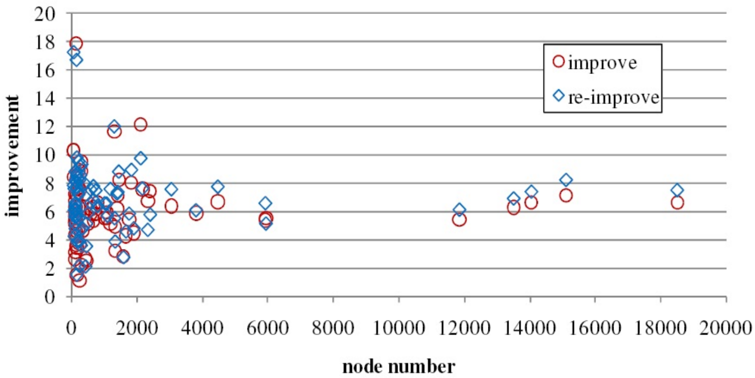

Fig. 1 The online improvements of the initial approximated solutions for the instance networks

largest improvement is 17.8157 for the instance network pr136; however, the best competitive ratio is 1.1644 for st70. Also, for reverse direction the best values of the improvement and the competitive ratio are respectively 17.2178 and 1.1417 both for berlin52.

6 Conclusions

This study considered the online symmetric travelling salesman problem. A discrete time Markov chain with uniformly distributed transition probabilities is established. Then, we applied the simulated annealing heuristic to improve the computational capability of the algorithm. Other stochastic models and decision criterion could be studied for future works.

References

1. G. Ausiello, V. Bonifaci, L. Laura, The on-line asymmetric traveling salesman problem, J. Discrete Algorithms, 6, 290-298 (2008).
2. G. Ausiello, M. Demange, L. Laura, V. Paschos, Algorithms for the on-line quota traveling salesman problem, Inf. Processing Letters, 92, 89-94 (2004).
3. N. Christofides, Worst-case analysis of a new heuristic for the travelling salesman problem, Carnegie Mellon University, Technical Report, CS-93-13 (1976).
4. G. Gutin, A.P. Punnen, The traveling salesman problem and its variations, Kluwer Academic Publishers, Boston (2004).
5. O.C. Ibe, Markov Processes for Stochastic Modeling, Academic Press, Boston (2009).
6. P. Jaillet, X. Lu, Online traveling salesman problems with service flexibility, Networks, 58, 137-146 (2011).
7. M.W. Park, Y.D. Kim, A systematic procedure for setting parameters in simulated annealing algorithms, Computers and Operations Res., 25, 207-217 (1998).
8. C.H. Papadimitriouc, J.N. Tsitsiklis, The complexity of Markov chain decision processes, Mathematics of Operation Res., 12, 441-450 (1987).
9. A. Ptrowski, J. Dro, E.P.S. Taillard, Metaheuristics for Hard Optimization, Springer, Heidelberg (2006).

10. G.H. Polychronopoulos, J.N. Tsitsiklis, Stochastic shortest path problems with recourse, Networks, 27, 133-143 (1996).
11. X. Wen, Y. Xu, H. Zhang, Online traveling salesman problem with deadline and advanced information, Computers and Industrial Engineering, 63, 1048-1053 (2012).
12. W. Yu, Z. Liu, X. Bao, Optimal deterministic algorithms for some variants of online quota traveling salesman problem, European J. Operational Res., 238, 735-740 (2014).
13. H. Zhang, W. Tong, Y. Xu, G. Lin, The steiner traveling salesman problem with online edge blockages, European J. Operational Res., 243, 30-40 (2015).
14. Ruprecht-Karls-Universitt Heidelberg, http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/index.html