Hindawi Publishing Corporation
Journal of Industrial Engineering
Volume 2013, Article ID 347825, 5 pages
http://dx.doi.org/10.1155/2013/347825

Research Article

A Novel Metaheuristic for Travelling Salesman Problem

Vahid Zharfi and Abolfazl Mirzazadeh

Industrial Engineering Department, Faculty of Engineering, Kharazmi University, Tehran 31979-37551, Iran

Correspondence should be addressed to Vahid Zharfi; v.zharfi@tmu.ac.ir

Received 30 April 2013; Accepted 3 July 2013

Academic Editor: Alan Chan

One of the well-known combinatorial optimization problems is travelling salesman problem (TSP). This problem is in the fields of logistics, transportation, and distribution. TSP is among the NP-hard problems, and many different metaheuristics are used to solve this problem in an acceptable time especially when the number of cities is high. In this paper, a new meta-heuristic is proposed to solve TSP which is based on new insight into network routing problems.

1. Introduction

Although there are various classic methods for solving optimization problems, but they are not always able to solve real and applied optimization problems. It is generally believed that these problems are known as difficult and complicated problems. TSP is a typical example of a very hard combinatorial optimization problem. TSP can be modelled as an undirected weighted graph, such that cities are the graph's vertices, paths are the graph's edges, and a path's distance is the edge's length. It is a minimization problem starting and finishing at a specified vertex after having visited each other's vertex exactly once. Metaheuristics can solve high-dimensional problems quickly. Heuristics may be classified into two families: specific heuristics and metaheuristics. Specific heuristics are tailored and designed to solve a specific problem and/or instance. Metaheuristics are general-purpose algorithms that can be applied to solve almost any optimization problem. They may be viewed as upper level general methodologies that can be used as a guiding strategy in designing underlying heuristics to solve specific optimization problems [1]. The word metaheuristic was first used by Glover [2] during introducing tabu search concepts.

Metaheuristics gained more popularity in few recent decades, and various algorithms are proposed to solve the problem such as Genetic Algorithm [3–5], Ant Colony [6–10], Particle Swarm [11], and Simulated Annealing [12, 13].

In this paper, a novel metaheuristic is proposed for solving TSP. This algorithm is used to foresee the best next city so as to gain shorter rout. The 2-Opt and 3-Opt algorithms are also used in local search. They are probably the most basic local search heuristics for the TSP.

The paper is organized as follows: Section 2 illustrates the background theory of suggested algorithm. In Section 3, formulating algorithms are presented. Algorithm improvement by using k-Opt is discussed in Section 4. The algorithm is employed into several standard TSP instances in the next section and the results are reported. According to the obtained results, it is noticeable that the algorithm achieves proper answers, quickly so that the answer of the first iteration will be a good and acceptable answer.

Finally, Section 6 concludes the paper.

2. Background Theory of the Algorithm

In this model, it is supposed that every node in the network is a vibration source and has a special weight which vibrates the edges, so that the nodes with the heavier weight release more vibration. This emitted force by a node distributes in edges that are connected to. Shared amount of each node is dependent on the edges distances. Shorter edges have more quota of this force. In other words, if the length of an edge connected to the node is shorter, less vibration falls down along the edge and it is vibrated more intensely. There is a simple network (Figure 1) to illustrate the theory. The

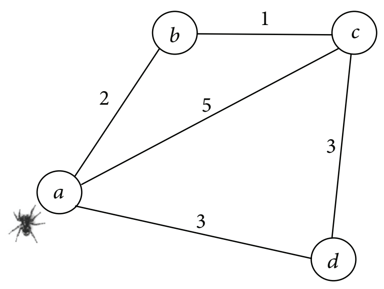

Figure 1: Routing in a network.

searcher is located at the node a in the following figure and wants to move to the next node after a.

Signs which lead to a are amount of vibrations that each edge indicates. Therefore, vibration of the edges can be considered as tendency that each neighbour node has in order to be selected by the searcher.

Node c as a sample has relationship with nodes a, b, and d. The distance for each edge is determined, and the sum of the inverted distances equals $(1/5 + 1 + 1/3) = 23/15$. Weight of this node is w_c. Therefore, the vibration that this node sends to the node a equals:

$$\frac{(1/5)}{(23/15)} \times w_c = 0.13 \times w_c. \quad (1)$$

Also, about nodes b and d, the amount of vibration which is sent to node a is, respectively,

$$\begin{aligned} &\frac{(1/2)}{(3/2)} \times w_b = 0.33 \times w_b, \\ &\frac{(1/3)}{(2/3)} \times w_d = 0.05 \times w_d. \end{aligned} \quad (2)$$

If all weights are considered equal to 1, the searcher receives a total vibration with an amount of $(0.13 + 0.33 + 0.50) = 0.96$ in node a. As a result, the searcher at the node a selects c with 14% possibility (0.13/0.96), b with 34% possibility (0.33/0.96), and d with 52% possibility (0.50/0.96).

Movement from node c in addition to direct route $c \rightarrow a$ can be transferred to node a of other routes such as "$c \rightarrow d \rightarrow a$" and "$c \rightarrow b \rightarrow a$." If we also consider binary routes, calculations would be changed. Thus, by considering routes with maximum two edges, there are three different routes from c to a (Table 1). Also, about nodes b and d, similar calculations will be performed (Tables 2 and 3).

According to the tables, it is possible to obtain an amount of vibration in each edge. For example, edge "$c \rightarrow a$" receives a force with an amount of $(0.20/2.28) \times w_c$ unit of c, $(0.16/2.25) \times w_b$ unit of b, and a force with an amount of $(0.12/1.35) \times w_d$ unit of node d. Therefore, "$c \rightarrow a$" edge vibrates as follows:

$$\left(\frac{0.20}{2.28}\right) \times w_c + \left(\frac{0.16}{2.25}\right) \times w_b + \left(\frac{0.12}{1.35}\right) \times w_d. \quad (3)$$

It is supposed that all weights are 1. Therefore, the amount of vibration in "$c \rightarrow a$" is 0.25. Similarly, "$b \rightarrow a$" and "$d \rightarrow a$" are 0.37 and 0.31, respectively. It should be mentioned that the total amount of these three values is not necessarily equal to 1. In the investigated example final total amount of vibration force of the three edges which end with a is 0.93.

Now, the searcher is faced with vibrant edges in node a Therefore, it selects "$b \rightarrow a$" with 40% possibility (0.40/0.93) and moves toward "b," selects "$c \rightarrow a$" with 27% possibility (0.25/0.93) and moves toward "c," or also selects "$d \rightarrow a$" with 33% possibility (0.31/0.93) and moves toward node "d." Total amount of these possibilities equals 1.

3. Model Formulation

In the previous section we got acquainted with the method of routing by the algorithm. As mentioned, edges with more vibration are selected with more possibility.

The vibration of a supposed edge "$j \rightarrow i$" is derived from two factors:

(i) direct force which is emitted from jth node;

(ii) indirect forces that are emitted from other nodes and distribute up to "$j \rightarrow i$." In other words, it is released from node k $(k \neq i, j)$ to node i after passing node j:

$$\begin{aligned} D_{kji} &= \begin{cases} d_{kj} + d_{ji} & \text{if } k \neq i, j, \\ d_{ji} & \text{if } k = j \neq i, \end{cases} \\ D_{kji} &= \frac{1}{d_{kji}}. \end{aligned} \quad (4)$$

In this relation d_{kji} is the distance of the route initiated from k that ends in i after passing j. Through definition D_k is sum of inverted distances of routes connected to k which has up to two edges (see (5)); it is possible to gain the force which is applied from node k to edge $j \rightarrow i$ by:

$$D_k = \sum_{i \neq k} \sum_{j \neq k} D_{kji}, \quad (5)$$

$$r_{kji} = \left(\frac{D_{kji}}{D_k}\right) \times w_k, \quad (6)$$

so that w_k represents weight of kth node. r_{kji} is the applied force for "$j \rightarrow i$" resulting from k. Vibration of "$j \rightarrow i$" equals

$$v_{ji} = \sum_k r_{kji}. \quad (7)$$

The final result of the previous calculations is matrix $V = (v_{ji})$ in which v_{ji} represents vibration degree in the edge "$j \rightarrow i$." It is necessary to use another element in order to increase ability of this matrix. To do so we can use vision power matrix.

In distance matrix, d_{ji} represents the length of an edge which is connector of nodes i and j. The vision matrix $\Gamma = (\gamma_{ij})$ is defined as follows:

$$\gamma_{ij} = \frac{1}{d_{ij}}. \quad (8)$$

Table 1: Emitted force by node "c" in the network.

Origin	Destination	Route	Route distance	Inversion of route distance	Final edge of the route	Amount of vibration in the final edge
c	a	$c \to b \to a$	3	0.33	$b \to a$	$(0.33/2.28) \times w_c$
c	a	$c \to a$	5	0.20	$c \to a$	$(0.20/2.28) \times w_c$
c	a	$c \to d \to a$	6	0.16	$d \to a$	$(0.16/2.28) \times w_c$
c	b	$c \to a \to b$	7	0.14	$a \to b$	$(0.14/2.28) \times w_c$
c	b	$c \to b$	1	1.00	$c \to b$	$(1.00/2.28) \times w_c$
c	d	$c \to a \to d$	8	0.12	$a \to d$	$(0.12/2.28) \times w_c$
c	d	$c \to d$	3	0.33	$c \to d$	$(0.33/2.28) \times w_c$
	Total			2.28	w_c	

Table 2: Emitted force by node "b" in the network.

Origin	Destination	Route	Route distance	Inversion of route distance	Final edge of the route	Vibration force in the final edge
b	a	$b \to a$	2	0.50	$b \to a$	$(0.50/2.25) \times w_b$
b	a	$b \to c \to a$	6	0.16	$c \to a$	$(0.16/2.25) \times w_b$
b	c	$b \to a \to c$	7	0.14	$a \to c$	$(0.14/2.25) \times w_b$
b	c	$b \to c$	1	1.00	$b \to c$	$(1.00/2.25) \times w_b$
b	d	$b \to a \to d$	5	0.20	$a \to d$	$(0.20/2.25) \times w_b$
b	d	$b \to c \to d$	4	0.25	$c \to d$	$(0.25/2.25) \times w_b$
	Total			2.25		w_b

By combination of two matrixes Γ and V, it is possible to use both visual sense and mechanical sensor of the searcher for detection and recognition of the best route.

The result of performed calculations in this section is possibility matrix $P = (p_{ij})$:

$$P_{ij} = \frac{v_{ij}^{\alpha} \cdot \gamma_{ij}^{\beta}}{\sum_i \sum_j v_{ij}^{\alpha} \cdot \gamma_{ij}^{\beta}}. \quad (9)$$

The parameters $\propto$ and β are investigated from the obtained results. A proper amount of these parameters is dependent on nature and structure of the investigated problem. Adjustment of $\propto$ and β has an important role in algorithm efficiency. Investigation of the obtained results shows that $\propto = 8$ and $\beta = 2$ are appropriate for most of the instances.

4. Algorithm Improvement by Local Search

The suggested algorithm searches to get to the best route in the network globally. But it makes less chance to envelop all answers. Consequently, local search is necessary in order to cover the solution space more entirely. A local search algorithm starts from a candidate solution and then iteratively moves to a neighbour solution.

In optimization, 2-Opt and 3-Opt are simple local search algorithms for solving the travelling salesman problem [14, 15]. In this paper, we propose to combine the 2-Opt, 3-Opt, and the suggested algorithm to improve the answers. The algorithm steps are as follows:

```
Initialize matrix $P$
**For** (i < Iteration Max) do:
  **While** the searcher complete a solution
      Choose probabilistically the next state to move;
      Add that move to the tabu list;
  **end**
  Implementation of the 2-Opt and 3-Opt;
  **If** (local best solution better than global solution)
    Save local best solution as global solution;
  **end**
  Update matrix $P$ for each edge that the searcher chose
  (Increase elements of matrix $P$ related to the selected
  route up to $q$ percent);
**end**
```

5. Computational Results

In this section we are going to solve different standard examples by using the proposed algorithm. A large number of standard samples are available in public references. These references include famous samples for combinatorial optimization problems. One of the valid references in travelling salesman problems is TSPLIB [16].

5.1. Quality of Solutions. In order to estimate the algorithm function from absolute difference obtained by optimized answer, the following relation is used:

$$\frac{|f(s) - f(s^*)|}{f(s^*)}, \quad (10)$$

so that s is the obtained answer and s^* is the optimized answer.

Table 3: Emitted force by node "d" in the network.

Origin	Destination	Route	Route distance	Inversion of route distance	Final edge of the route	Vibration force in the final edge
d	a	$d \to c \to a$	8	0.12	$c \to a$	$(0.12/1.35) \times w_d$
d	a	$d \to a$	3	0.33	$d \to a$	$(0.33/1.35) \times w_d$
d	b	$d \to a \to b$	5	0.20	$a \to b$	$(0.20/1.35) \times w_d$
d	b	$d \to c \to b$	4	0.25	$c \to b$	$(0.25/1.35) \times w_d$
d	c	$d \to a \to c$	8	0.12	$a \to c$	$(0.12/1.35) \times w_d$
d	c	$d \to c$	3	0.33	$d \to c$	$(0.33/1.35) \times w_d$
	Total			1.35		w_d

Table 4: Searching for optimized solution of different travelling salesman problems.

Instance	Best known	First sol.	Best sol.	Average sol.	Worst sol.	Error
ftv33	1286	1462	1286	1286	1286	0.0000000
ftv38	1530	1591	1530	1535	1546	0.0032679
swiss42	1273	1364	1273	1273	1273	0.0000000
ftv44	1613	1742	1615	1619	1625	0.0037197
ry48p	14422	15341	14466	14493	14507	0.0049230
berlin52	7542	7951	7542	7542	7542	0.0000000
ftv64	1839	2022	1909	1925	1940	0.0467645
eil76	538	559	538	538	538	0.0000000
eil101	629	717	630	638	650	0.0143084

5.2. Robustness. Metaheuristics must be able to act properly in an extended range of samples or problems with the same parameters. It is possible to adjust metaheuristic parameters for a collection of samples in a good way which has less efficiency for other samples.

Different problems of travelling salesman have been solved with different sizes and structures by the algorithm. For all instances, 9, 2, and 0.2% are allocated for $\propto$, β, and q, respectively (Table 4).

The optimal path length of the algorithm's iterations for berlin52 is shown in Figure 2.

According to the obtained results, it is noticeable that the algorithm achieves proper answers quickly so that the answer of the first iteration will be a good and acceptable answer. Accordingly, although the suggested algorithm shows its success alone in this issue, it can be used as a powerful initial solution generator for other solution-based metaheuristics in order to use advantages of other algorithms and getting more close to the optimized answer.

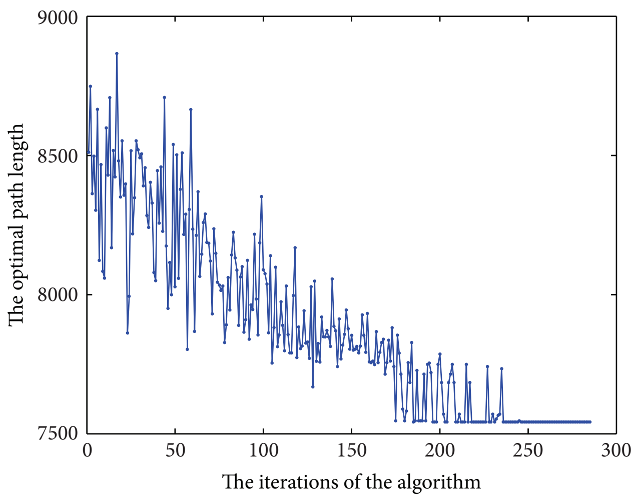

Figure 2: The optimal path length of proposed algorithm for berlin52.

6. Conclusion

In this paper we presented a new metaheuristic that has new insight into network problems. At the end of every iteration, the algorithm also tries to improve both the performance of the algorithm and the quality of solutions for the Tsp by using local search algorithms in the name of 2-Opt and 3-Opt. As a result, the proposed algorithm can be used for variety network problems due to its insight to these kinds of problems. Therefore, through extending this behaviour we can move toward solving different optimization problems by this algorithm in a way that each position of problem solutions represents a node and each algorithm step is passing from one state to another state. Future work will be conducted to improve the proposed algorithm and examine other local search algorithms to enhance the performance of the method for improving solutions. Additional improvements might lie on the combination of our approach with the other metaheuristics like genetic algorithm, tabu search, simulated annealing, and so on.

References

[1] T. El-ghazali, *Meta-Heuristic from Design to Implementation*, Wiley, 2009.

[2] F. Glover, "Future paths for integer programming and links to artificial intelligence," *Computers and Operations Research*, vol. 13, no. 5, pp. 533–549, 1986.

[3] S. Yuan, B. Skinner, S. Huang, and D. Liu, "A new crossover approach for solving the multiple travelling salesmen problem using genetic algorithms," *European Journal of Operational Research*, vol. 228, no. 1, pp. 72–82, 2013.

[4] Y. Nagata and D. Soler, "A new genetic algorithm for the asymmetric traveling salesman problem," *Expert Systems with Applications*, vol. 39, no. 10, pp. 8947–8953, 2012.

[5] W. Hui, "Comparison of several intelligent algorithms for solving TSP problem in industrial engineering," *Systems Engineering Procedia*, vol. 4, pp. 226–2235, 2012.

[6] S. M. Chen and C. Y. Chien, "Solving the traveling salesman problem based on the genetic simulated annealing ant colony system with particle swarm optimization techniques," *Expert Systems with Applications*, vol. 38, no. 12, pp. 14439–14450, 2011.

[7] J. Yang, X. Shi, M. Marchese, and Y. Liang, "Ant colony optimization method for generalized TSP problem," *Progress in Natural Science*, vol. 18, no. 11, pp. 1417–1422, 2008.

[8] M. Dorigo and L. M. Gambardella, "Ant colonies for the travelling salesman problem," *BioSystems*, vol. 43, no. 2, pp. 73–81, 1997.

[9] J. Bai, G. K. Yang, Y. W. Chen, L. S. Hu, and C. C. Pan, "A model induced max-min ant colony optimization for asymmetric traveling salesman problem," *Applied Soft Computing*, vol. 13, no. 3, pp. 1365–1375, 2013.

[10] A. Colorni, M. Dorigo, F. Maffioli, V. Maniezzo, G. Righini, and M. Trubian, "Heuristics from nature for hard combinatorial optimization problems," *International Transactions in Operational Research*, vol. 3, no. 1, pp. 1–21, 1996.

[11] X. H. Shi, Y. C. Liang, H. P. Lee, C. Lu, and Q. X. Wang, "Particle swarm optimization-based algorithms for TSP and generalized TSP," *Information Processing Letters*, vol. 103, no. 5, pp. 169–176, 2007.

[12] K. Meer, "Simulated annealing versus metropolis for a TSP instance," *Information Processing Letters*, vol. 104, no. 6, pp. 216–219, 2007.

[13] X. Geng, Z. Chen, W. Yang, D. Shi, and K. Zhao, "Solving the traveling salesman problem based on an adaptive simulated annealing algorithm with greedy search," *Applied Soft Computing Journal*, vol. 11, no. 4, pp. 3680–3689, 2011.

[14] C. Engels and B. Manthey, "Average-case approximation ratio of the 2-opt algorithm for the TSP," *Operations Research Letters*, vol. 37, no. 2, pp. 83–84, 2009.

[15] K. H. Hsieh, "Fourier descriptors for 2-Opt and 3-Opt heuristics for traveling salesman problem," *Journal of the Chinese Institute of Industrial Engineers*, vol. 28, no. 3, pp. 237–246, 2011.

[16] http://elib.zib.de/pub/Packages/mp-testdata/tsp/tsplib/tsplib .html.