

## Integer Programming Model for Two-Centered Double Traveling Salesman Problem

**Mehmet Fatih Demiral**

Assist. Prof.Dr., Department of Industrial Engineering, Faculty of Engineering and  
Architecture Mehmet Akif Ersoy University, Burdur, Turkey

**Halil Şen**

Assist. Prof.Dr., Department of Industrial Engineering, Faculty of Engineering and  
Architecture Mehmet Akif Ersoy University, Burdur, Turkey

### Abstract

Traveling Salesman Problem (TSP) is among the most popular combinatorial problems and has been widely studied with many extensions in the literature. There have been integer programming formulations and solution approaches for TSP and its variations. One of the most popular topics is the multiple TSP (m-TSP). It has been started to work on the last decades. Generally, m-TSP has a single depot and more than one tour. However, some types have more than one depot. Besides, if seeking, many encounter with double traveling salesman problem (d-TSP). As inferred from the literature, d-TSP is a variation of m-TSP in which two salespersons operate in parallel. They start and end either in a single or two depots. Apart from the literature, a new TSP model has been constructed in this study. The model analyzes the behaviors of traveling salesmen both in the main and secondary tour. Tours have been aggregated via single node existing in the secondary tour. In the application part, Simulated Annealing (SA) was used to optimize the traveling paths of salesmen. The objective parameters of both tours have been explored within a numerical example. According to them, several cost values have been found. The optimum parameters and costs of both tours are to be determined and some practical issues relevant to the behaviors of traveling salesmen have been given. Results suggest that alternative travel plans of traveling salesmen could be possible. Also, findings give hints about both tours and their characteristics.

**Keywords:** Double Traveling Salesman Problem, Integer Programming, Combinatorial Optimization.

### 1. Introduction

The traveling salesman problem is among the popular topics in the field of combinatorial optimization. The salesman has an objective to minimize the travel distance by visiting each city only once. The complexity of TSP grows exponentially as number of nodes increases. If huge number of nodes exist, TSP gets substantially hard. Therefore, new techniques and methodologies have been developed so as to solve TSP effectively. Heuristic and metaheuristic solution procedures are the most accepted ones to cope with TSPs. TSP covers many sub-topics which have close relations with other complex structured routing problems. By the way, TSP (decision version) is NP-complete problem. TSP is also in class NP and many other problems, such as satisfiability problem (SAT), clique problem, vertex cover problem and hamiltonian path problem reduce to TSP decision problem (Farahani & Hekmatfar (eds.), 2009, p. 33-34). Basically, TSP has two widely researched variants, named as traveling purchaser problem (TPP) and vehicle routing problem (VRP). Those are applicable and useful for operations research and especially in the field of many industries. Other than TPP and VRP, m-TSP is another form of TSP in which more than one salesman operates through multiple paths. Bektas (2006) gives some formulations and solution procedures for m-TSP. Although there exist quite more publications upon TSP and VRP, m-TSP has less attention in the literature. Possible variations of m-TSP are classified by number of depots, number of salesmen, the number of salesmen having fixed charges, time windows and the other special constraints. One of the most widely studied m-TSP is double traveling salesman problem (d-TSP). In d-TSP, some types have two depots and two salesmen operate independently. Other types have single depot and they return to the original depot. If the salesmen being used in optimization are not stated, then each salesman usually has an associated fixed cost incurring whenever this salesman used in this solution. On the other hand, number of tours is fixed in d-TSP. D-TSP has two tours in all cases.

In this research, a new model has been proposed for d-TSP, named as two centered d-TSP (tcd-TSP). The reason of using this term is the existing of two depots (centers) which are waiting to be selected among other nodes. One of them is the home depot of primary tour and the other is the home depot of secondary tour. Therewithal, the second center is the aggregation point of both tours. To be clear, the model investigates the relative status of primary and secondary paths. Hence, two parameters are set to evaluate both tours and their characteristics. Eventually, the optimum traveling plan and the optimum parameters for tcd-TSP are to be found out.

## 2. Literature Research

There has been wide range of publication upon TSP and its variations. Burkard, Deineko, van Dal, van der Veen and Woeginger (1998) write a survey upon special cases of the TSP. They mention about symmetric and asymmetric matrices, euclidean TSP and the various special cases of the TSP. One of the leading classifications is depending on data whether to be symmetric or asymmetric. In symmetric TSP (s-TSP), the cost measure  $d_{ij} = d_{ji}$  associated with edge  $(i, j) \in A$  is valid for all pair of points. If  $d_{ij} \neq d_{ji}$  for at least one, then the TSP becomes asymmetric TSP (a-TSP). Several a-TSP formulations are given in (Oncan, Altinel & Laporte, 2009). There have been recently found many publications on the two types of TSP worked out through heuristics. To solve the well-known symmetric and asymmetric TSPs, a new version of bat algorithm is introduced in (Osaba, Yang, Diaz, Lopez-Garcia & Carbelledo, 2016). Brest and Zerovnik (2005) developed a heuristic algorithm for the a-TSP and studied with instances from TSPLIB library. A new genetic algorithm for asymmetric version is proposed and solved on benchmark instances in (Nagata & Soler, 2012). The other widely studied classification is depending on the number of tours in the problem. The number of tours can be single, double or multiple. In this context, m-TSP can be defined as in (Bektas, 2006, p. 209-210): In a given of nodes, let there are  $m$  salesmen located at a single depot node. The remaining nodes that are to be visited are intermediate nodes. Then the m-TSP consists of finding tours for all  $m$  salesmen, who all start and end at the depot such that each intermediate node is visited exactly once and the total cost of visiting all nodes is minimized. The cost metric can be defined in terms of distance, time etc. The basic extensions of m-TSP are also touched upon in this section. Since TSP is a special version of m-TSP, if one tour exists, m-TSP reduces to the TSP. If double tour exists, m-TSP reduces to the d-TSP. Likewise, the other versions of m-TSP can be obtained as the number of tours in the problem changes. There has been limited number of publication upon m-TSP and its extensions. In this study, some of the important ones are given. Kara and Bektas (2006) give some integer programming formulations for m-TSP and its variations. They show that the solution of multidepot m-TSP with the proposed formulation is significantly superior to previous approaches. Cheang, Gao, Lim, Qin and Zhu (2012) extend the TSP with pick-up delivery and LIFO loading (TSPPDL) by considering two additional factors, the use of multiple vehicles and a limitation on the total distance that a vehicle can travel. Chan and Baker (2005) consider the multiple depots multiple TSFLP expanded to include vehicle range and multiple service frequency requirements. Artificial intelligence has started to play an important role in solving m-TSP. By using genetic algorithm and evolutionary programming, a new methodology for m-TSP was developed in several studies (Carter & Ragsdale, 2006; Kota & Jarmai, 2015; Yuan, Skinner, Huang & Liu, 2013). Modares, Somhom and Enkawa (1999) propose an algorithm for TSP based on self organizing neural network approach and extend it to m-TSP. In the following researches (Mladenovic, Urosevic, Hanafi & Ilic, 2012; Soylu, 2015), VNS approach was used to solve m-TSP. As mentioned before, one of the important extensions of m-TSP is the d-TSP. Plebe and Anile (2002) developed an algorithm and applied a neural network based approach to d-TSP. One of the most common research topics of d-TSP is the d-TSP with multiple stacks (d-TSPMS). There have been quite a lot of publications in this version. Petersen and Madsen (2009) introduce the problem and present four different metaheuristic approaches to its solution. Efficient algorithms for the problem are presented in (Cazassa, Ceselli & Nunkesser, 2012) and exact methods for the problem are developed in (Iori & Ledesma, 2015). Petersen, Archetti and Sprezza (2010) present mathematical formulations and solution approaches for the problem. It can be counted more publications in relation to d-TSP and its extensions.

The remainder of the paper is organized as follows: Section 3 formally introduces the tcd-TSP and gives the integer programming formulation. In Section 4, a simulated annealing based algorithm for the problem is proposed and the model is tested for a group of parameters and findings are given. Finally, Section 5 summarizes and comments on the contribution of this paper.

## 3. Mathematical Model for tcd-TSP

As mentioned in introduction part, a new mathematical model has been developed in this section. In tcd-TSP, the problem has two center points selecting among nodes and tours are aggregated via single node existing in the secondary tour. The

center points are represented as the home depots of the tours. Thus, the optimum solution changes as the home depots change. In addition, each travel path is set by a parameter. Thence, the optimum solution is also affected by the objective parameters. In light of this information, the basic tcd-TSP can be formulated as follows:

**Description of Sets:**

$$N = \{1, 2, \dots, p_1, \dots, n\}$$

$$M = \{n + 1, n + 2, \dots, p_2, \dots, m\}$$

$$R = N \cup M$$

**Parameters:**

- $\alpha$  : cost parameter of primal tour
- $\beta$  : cost parameter of secondary tour
- $c_{ij}$  : distance value between city  $i$  and  $j$  in the primal tour
- $d_{ij}$  : distance value between city  $i$  and  $j$  in the secondary tour

**Decision Variables:**

$$x_{ij} = \begin{cases} 1, & \text{if salesman travels between city } i \text{ and } j \\ 0, & \text{otherwise} \end{cases}$$

$$\text{Min } Z = \alpha * \sum_{i,j \in N \cup \{p_2\}}^n \sum_{i,j \in N \cup \{p_2\}}^n c_{ij} * x_{ij} + \beta * \sum_{i,j \in M}^m \sum_{i,j \in M}^m d_{ij} * x_{ij}$$

s.t.

$$\sum_{j \in N \cup \{p_2\}}^n x_{ji} = 1 \quad \forall i \in N \cup \{p_2\} \tag{1}$$

$$\sum_{j \in N \cup \{p_2\}}^n x_{ij} = 1 \quad \forall i \in N \cup \{p_2\} \tag{2}$$

$$\sum_{j \in M}^m x_{ji} = 1 \quad \forall i \in M \tag{3}$$

$$\sum_{j \in M}^m x_{ij} = 1 \quad \forall i \in M \tag{4}$$

$$\sum_{i \in S_1} \sum_{j \notin S_1} x_{ij} \geq 1 \quad |S_1| \geq 2 \wedge S_1 \subset N \cup \{p_2\} \tag{5}$$

$$\sum_{i \in S_2} \sum_{j \notin S_2} x_{ij} \geq 1 \quad |S_2| \geq 2 \wedge S_2 \subset M. \tag{6}$$

$$\forall x_{ij} \in \{0, 1\} \quad \exists (i, j) \in N \cup \{p_2\} \wedge \exists (i, j) \in M \tag{7}$$

In the model formulation, objective function minimizes the total cost of two salesmen in terms of distance. Equations (Eqs.) (1) and (2) are the degree constraints for the primal tour. Eqs. (3) and (4) are the degree constraints for secondary tour. Inequalities (Ineq.) (5) and (6) are the subtour elimination constraints for tour 1 and 2, respectively. Expression (7) gives decision variables and states that some of the nodes belong to primary tour and the remaining belongs to the other one. The integer programming formulation has  $n^2+n+m^2-m$  zero-one decision variables and  $2^{n+1}+2^m+n+m$  constraints for each model constructed because of the unknown location of transfer point. In total, there exist  $mn^2+m^3+mn-m^2$  zero-one decision variables and  $m2^{n+1}+m2^m+mn+m^2$  constraints in the integer programming model. Thus, the formulation has  $O(mn^2)$  integer variables and  $O(m2^{n+1})$  constraints if inequality  $n \geq m$  is hold. The tcd-TSP formulation has approximately the four thousand variables and twenty million inequalities when  $n=20, m=10$ .

The proposed integer programming model for tcd-TSP is convenient for tourism organizations, cargo and freight transportation by all modes, the design of two-piece objects, fixing the transfer/transshipment points and depot locations and etc.

#### 4. Application of the SA Algorithm for the tcd-TSP

Kirkpatrick, Gelatt, and Vecchi (1983) introduced the simulated annealing algorithm which attempts to solve hard combinatorial optimization problems in operational research area by using the annealing process of hard metals. The authors found out the relation between the annealing process and optimization. Thereafter, they formulated the relation in compact form and it has begun to apply in many optimization problems. In this section, a neighborhood structure has been developed and embedded in SA algorithm. Here, also a distinct objective function is evaluated for two interrelated tours. This structure is peculiar to tcd-TSP. As explained clearly, the overall steps of SA are given below:

##### 4.1. Steps of the SA Algorithm for tcd-TSP

*Step 1.* Select the initial tours A and B, objective parameters  $\alpha$  and  $\beta$ , iteration limit K, the iteration limit L for temperature change, initial temperature  $T_0$ , cooling rate  $r$  ( $0 < r < 1$ ). Global solution  $A^* = A^1$ ,  $B^* = B^1$ , global index  $k=1$  and iteration counter  $l=1$ .

*Step 2.* Select a point randomly from secondary tour B and fix it as transfer point between A and B. Revise the primal tour A including the home depot location.

*Step 3.* Define two indices for both tours separately and exchange the points represented by them. For instance, indices a, b for A and indices d, e for B. Calculate the net improvement as  $\Delta Z = \left[ Z(\alpha^* A^k + \beta^* B^k) - Z(\alpha^* A^{k+1} + \beta^* B^{k+1}) \right]$ . It should be here noticed that alpha and beta can get out of the brackets since they are parameters.

*Step 4.* If  $\Delta Z \geq 0$  (min.), then accept the solution and update it. Go to step 6, otherwise go to step 5.

*Step 5.* If  $P = \exp\left[-\frac{Z(\alpha^* A^{k+1} + \beta^* B^{k+1}) - Z(\alpha^* A^k + \beta^* B^k)}{T_k}\right] \geq 0.5$  then update  $A^{k+1} = A^k + \Delta A^{k+1}$  and  $B^{k+1} = B^k + \Delta B^{k+1}$ , otherwise update  $A^{k+1} = A^k$  and  $B^{k+1} = B^k$ . Go to step 7.

*Step 6.* If the objective value  $Z(\alpha^* A^{k+1} + \beta^* B^{k+1})$  is superior to global objective value, update the global solutions  $A^* = A^{k+1}$  and  $B^* = B^{k+1}$ .

*Step 7.*  $k = k + 1, l = l + 1$ . If  $l = L$  take  $l = 1$  and update  $T$  as  $T = r * T$ . Return to Step 2.

*Step 8.* If  $k = K$  then stop the algorithm.

##### 4.2. Data and Parameter Settings

In this application, a kind of medium size problem has been handled with size ( $n=10, m=5$ ). The random data has been used in the application. In order not to converge quickly the poor solutions, a relatively high initial temperature  $T_0$  has been taken ( $T_0=100000$ ) in SA algorithm. Cooling rate is also taken high because of keeping the accepting probability of poor results high ( $r=0.80$ ).  $L=1500$ , the iteration limit for temperature change is sufficient to get optimum results for this medium size problem.  $K=35000$ , the global iteration limit is also sufficient to get best results for the problem ( $n=10, m=5$ ).

##### 4.3. Computational Results

The SA algorithm for the tcd-TSP was developed using MatlabR2015b and run on a PC Pentium (R) Dual-Core CPU 3.33 GHz speed with 4 GB of RAM. The proposed SA based algorithm converged to the optimal solution within acceptable time. The coded algorithm was run 10 times for each pair of alpha and beta and 35000 iterations for each run. As seen in Table 1,  $\alpha$  varies in the interval [10,100] and  $\beta$  is kept fixed at 0.01. In Table 2,  $\alpha$  is kept fixed at 90 and  $\beta$  varies in the interval [0.01, 1]. Although the algorithm explores the solution according to the objective function, the optimum selection may be done after all of the algorithmic processes are completed. In the study, optimum traveling path of salesmen has been selected from Table 1 and 2 according to the sum of the primary and secondary tour lengths. Here, the findings indicate that the objective function of model provides the alternative solutions to the tcd-TSP. However, the optimum has been

determined among the alternative solutions. The proposed SA based algorithm was found successful to get acceptable results for this problem. Alternatively, many neighborhood structures can be developed and used successfully. Table 1 tells that the tour number nine is optimal and Table 2 shows that again, the same tour is optimal when the total tour length is considered. Besides this information, optimal parameter values have been found in the interval for alpha and beta. The optimal traveling path of salesmen and its length are as follows.

The Primary Tour

**3-6-1-2-9-10-7-12-8-5-4-3**

Optimal Length of A: 24136, Alpha for A: 90

The Secondary Tour

**12-13-15-14-11-12**

Optimal Length of B: 2182, Beta for B: 0.01

Table 1: Alternative paths found according to alpha [10,100] and beta 0.01.

Path Number	Primary Tour (A)	Secondary Tour (B)	Total Length of Path	Alpha ( $\alpha$ )	Beta ( $\beta$ )	Objective Value
1	24136	3096	27232	10	0.01	241391
2	24136	3517	27653	20	0.01	482755
3	24136	4730	28866	30	0.01	724127
4	24136	4456	28592	40	0.01	965485
5	24136	2822	26958	50	0.01	1206828
6	24176	2182	26358	60	0.01	1450581
7	24136	2822	26958	70	0.01	1689548
8	24220	3517	27737	80	0.01	1937635
9	24136	2182	26318	90	0.01	2172261
10	24136	4456	28592	100	0.01	2413645

Source: Author's calculations using Matlab.

Table 2: Alternative paths found according to alpha 90 and beta [0.01,1].

Path Number	Primary Tour (A)	Secondary Tour (B)	Total Length of Path	Alpha ( $\alpha$ )	Beta ( $\beta$ )	Objective Value
1	24136	2182	26318	90	0.01	2172261
2	24136	3096	27232	90	0.02	2172302
3	24136	3096	27232	90	0.03	2172333
4	24136	3517	27653	90	0.05	2172416
5	24136	2822	26958	90	0.08	2172466
6	24136	2798	26934	90	0.1	2172520
7	24136	4035	28171	90	0.3	2173451
8	24136	3517	27653	90	0.5	2174000
9	24136	2798	26934	90	0.8	2174478
10	24136	3517	27653	90	1	2175757

**5. Conclusions**

In this paper, an integer programming model has been proposed for the tcd-TSP. The paper also introduces the SA based approach for this problem. In order to verify the validity and effectiveness of the proposed model and algorithm, the problem was solved through random data. The findings indicate that many alternative solutions can be obtained by using different pairs of alpha and beta. As observed in many tours, the length of second tour changes because of the relative size of parameters and the data used in the study. Thus, the results reveal that the cost of path changes as the preferences of traveling salesmen on the path change. Therewithal, the model ensures to fix and update the optimum location of depots.

The proposed model can be applied to various industrial sectors as transportation and tourism are the foremost. It helps the manager to control the travel plan whenever the cost of it changes. Further, the model specifies the optimum location

of depots and transfer points. However, the board of management may decide to move the optimum locations while considering the real world constraints. Hence, they apply the sub-optimal results.

As distinct from this study, new mathematical models which take the fixed and variable delivery costs into consideration will be proposed for the tcd-TSP. The new constraints, like time windows, can also be modeled within the problem. In addition to that, new algorithmic approaches will be used to solve more complex structured tcd-TSP. The developing ideas, mathematical models and algorithms upon this problem will be applied to real world problems.

## References

- [1] Bektas, T. (2006). The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, 34, 209-219.
- [2] Brest, J. & Zerovnik, J. (2005). Proceedings from MIC2005: *The 6th Metaheuristics International Conference*. Vienna, Austria.
- [3] Burkard, R. E., Deineko, V. G., van Dal, R. van der Veen, J. A. A. & Woeginger, G. J. (2016). Well-solvable special cases of the traveling salesman problem: a survey. *SIAM Review*, 40(3), 496-546.
- [4] Carter, A. E. & Ragsdale, C. T. (2006). A new approach to solving the multiple traveling salesperson problem using genetic algorithms. *European Journal of Operational Research*, 175, 246–257.
- [5] Cazassa, M., Ceselli, A. & Nunkesser, M. (2012). Efficient algorithms for the double traveling salesman problem with multiple stacks. *Computers & Operations Research*, 39, 1044–1053.
- [6] Chan, Y. & Baker, S.F. (2005). The multiple depot, multiple traveling salesmen facility-location problem: vehicle range, service frequency, and heuristic implementations. *Mathematical and Computer Modeling*, 41, 1035-1053.
- [7] Cheang, B., Gao, X., Lim, A., Qin, H. & Zhu, W. (2012). Multiple pickup and delivery traveling salesman problem with last-in-first-out loading and distance constraints. *European Journal of Operational Research*, 223, 60-75.
- [8] Farahani, R. Z. & Hekmatfar, M. (Eds.). 2009. *Facility location: concepts, models, algorithms and case studies, contributions to management science*, Heidelberg, BW: Physica-Verlag.
- [9] Iori, M. & Riera-Ledesma, J. (2015). Exact algorithms for the double vehicle routing problem with multiple stacks. *Computers & Operations Research*, 63, 83–101.
- [10] Kara, I. & Bektas, T. (2006). Integer linear programming formulations of multiple salesman problems and its variations. *European Journal of Operational Research*, 174, 1449-1458.
- [11] Kirkpatrick, S., Gelatt, C., & Vecchi, M. (1983). Optimization by simulated annealing. *Science*, 220, (4598), 671–680.
- [12] Kota, L. & Jarmai, K. (2015). Mathematical modeling of multiple tour multiple traveling salesman problem using evolutionary programming. *Applied Mathematical Modeling*, 39, 3410–3433.
- [13] Mladenovic, N., Urosevic, D., Hanafi, S. & Ilic, A. (2012). A general variable neighborhood search for the one commodity pickup-and-delivery travelling salesman problem. *European Journal of Operational Research*, 220, 270–285.
- [14] Modares, A., Somhom, S., & Enkawa, T. (1999). A self-organizing neural network approach for multiple traveling salesman and vehicle routing problems. *International Transactions in Operational Research*, 6, 591-606.
- [15] Nagata, Y. & Soler, D. (2012). A new genetic algorithm for the asymmetric traveling salesman problem. *Expert Systems with Applications*, 39, 8947-8953.
- [16] Oncan, T., Altinel, I. K. & Laporte, G. (2009). A comparative analysis of several asymmetric traveling salesman problem formulations. *Computers & Operations Research*, 36, 637 – 654.
- [17] Osaba, E., Yang, X.-S., Diaz, F., Lopez-Garcia P. & Carbelledo, R. (2016). An improved discrete bat algorithm for symmetric and asymmetric traveling salesman problems. *Engineering Applications of Artificial Intelligence*, 48, 59-71.
- [18] Petersen, H. L., Archetti, C. & Sprenza, M. G. (2010). Exact solutions to the double travelling salesman problem with multiple stacks. *Networks*, 56(4), 229-243.

- [19] Petersen, H. L. & Madsen, O. B. G. (2009). The double travelling salesman problem with multiple stacks – Formulation and heuristic solution approaches. *European Journal of Operational Research*, 198, 139–147.
- [20] Plebe, A., & Anile, A. M. (2002). A neural network based approach to the double traveling salesman problem. *Neural Computation*, 14(2), 437-71.
- [21] Soyulu, B. (2015). A general variable neighborhood search heuristic for multiple traveling salesmen problem. *Computers & Industrial Engineering*, 90, 390–401.
- [22] Yuan, S., Skinner, B., Huang, S. & Liu, D. (2013). A new crossover approach for solving the multiple travelling salesmen problem using genetic algorithms. *European Journal of Operational Research*, 228, 72–82.