



Time-Windowed Vehicle Routing Problem: Tabu Search Algorithm Approach

Hasibe Berfu Demir^a, Ebru Pekel Özmen^b and Sakir Esnaf^c

^a Department of Industrial Engineering, Beykent University, Istanbul, Turkey

^b Department of Industrial Engineering, Samsun University, Samsun, Turkey

^c Department of Industrial Engineering, Istanbul-Cerrahpasa University, Istanbul, Turkey
pkl.ebru@gmail.com

KEYWORDS

vehicle routing problem;
tabu search optimization;
time window

ABSTRACT

Vehicle routing problem (VRP); it is defined as the problem of planning the best distribution or collection routes of the vehicles assigned to serve the scattered centers from one or more warehouses in order to meet the demands of the customers. Vehicle routing problem has been a kind of problem in which various studies have been done in recent years. Many vehicle routing problems include scheduling visits to customers who are available during certain time windows. These problems are known as vehicle routing problems with time windows (VRPTWs). In this study, a tabu search optimization is proposed for the solution of time window vehicle routing problem (VRPTWs). The results were compared with the current situation and the results were interpreted.

1. Introduction

Intense competition, products with short life curve, and increasing expectations of customers in today's global market have forced producers to invest in and give due consideration to distribution systems. This situation, with changes in communication and transportation technologies, has brought about the continuous development of logistic management (Bramel and Simchi-Levi, 1997). Logistic management is a crucial component of supply chain management. Solutions for daily encountered issues, regarding distribution, collection, or transportation of various goods, are searched for within the scope of logistic management. The main objective in logistic management is to minimize costs, and to fulfil needs expeditiously. The diversity and complexity of this field is reflected via the abundance of study areas, various methods and softwares (Crainic and Laporte, 1998).

Vehicle routing problems are placed in the center of distribution management. Every day, thousands of enterprises and organizations are to take care of the collection and delivery of various products, or

Hasibe Berfu Demir, Ebru Pekel Özmen,
Sakir Esnaf

Time-Windowed Vehicle Routing Problem:
Tabu Search Algorithm Approach



transportation of people from one place to another. As the constraints and purposes encountered in practice are quite flexible and distinctive, conditions within each business are different (Cordeau et al., 2007). To withstand the competition, decision makers in this field are to further rely on huge amounts of data, complex mathematical models and optimization techniques, as much as powerful computer and communication technologies.

2. Literature Review

Braysy and Gendreau (2002) analyze researches on Tabu Search methods for vehicle routing problem with time windows (VRPTW). VRPTW can be defined as the issue of designing minimum-cost routes for a vehicle fleet, from the main depot to spots geographically spread. Routes should be designed in such a manner that each spot will be visited only once, strictly by one vehicle, in a certain time period; and all routes begin and end at the depot. Moreover, the total number of demands of all the spots in a certain route should not exceed the vehicle's capacity. Apart from the designation of each method's basic characteristics, also presented and analyzed in the study are experimental results for Solomon's test problems. In their study, Ho and Haugland (2004) discuss vehicle routing problem with time windows and split delivery (VRPTWSD). They recommend a solving method based on tabu search for the problem which is taken without imposing constraints on split delivery options. Furthermore, analysis results of the practice for the discussed problem, carried out with 100 customers, are expressed clearly. In their study, Moccia and others (2012) explain a gradual tabu search method for generalized vehicle routing problems with time windows. The objective in this study is to provide a general tool that can be successfully applied to a wide range of specific problems. The algorithm is based on a tabu search heuristic method previously developed through the changing of neighborhood structure. In the study, experimental results are presented, and effectiveness of the approach is shown. Nguyen and others (2013) recommend a tabu search meta-intuitive method for multi-site vehicle routing problem with time windows. With the strategy which establishes control over two types of neighborhood relation which are equaled to two decision sets in certain stages of the problem, necessary tools for solving the problem are provided. Extensive numerical experiments, and comparisons with the literature, in the study, indicate that the suggested tabu search yields very high quality of solutions and improves those having been published recently. In their study, Gmira and others (2021) recommend a solution approach for vehicle routing problem with time windows in which travel speed is in relation with road sections in road network. This solution approach includes a tabu search intuitive method that focuses on shortest paths between two customers at different times of the day. The biggest contribution of this study is the development of techniques to evaluate in fixed time the applicability and approximate cost of the solution; and this way, the solution approach is enabled to take care of problematic examples possessing 200 nodes and 580 hops, in very acceptable calculation periods. The performance of the algorithm is evaluated through comparison with a full method in a series of comparative examples. Results indicate that high quality solutions are produced.

3. Methodology

3.1. Problem Definition

Vehicle routing problem can be defined as the distribution of product with different quantities to n number of customers by vehicles initially placed in the depot, and the collection of product from

customers. Designation of an optimum route to be used by vehicle-side while attempting to provide a group of users with service, presents a VRP problem. The purpose of the problem is to minimize the total cost of transportation. Solving the classical VRP problem includes determining a series of paths which start and finish at the depot, and which provide the constraint of giving service to each customer once. Classical vehicle routing problem (VRP) aims to find suitable routes with minimum costs (designating the shortest route, minimizing the number of vehicles, etc.). Some formulations also offer the shortening of maximum travel time (Belfiore et al., 2008). Vehicle routing problems are expressed as optimization problems occurring in the course of product distribution, the final stage of supply chain. And the approach taken into consideration to acquire the best solution, is the generating of constraints suitable for problem varieties. Some of the problem types discussed in VRP. Vehicle routing problem with time windows (VRPTW) is an improved version of classical vehicle routing problem with the addition of $[a,b]$ time period constraint (defined as time window) to each node. In a time window, (ai) states the earliest start time of service, and (bi) states the latest start time of service.

3.2. Problem constraints

The problem is modeled under the following assumptions:

- Routes must start and end at the warehouse within the time window of the warehouse.
- Each customer should be served with a single vehicle at a time.
- The total demand of customers on the same route should not exceed the vehicle capacity.
- Customers should serve within the time window. Early arriving vehicles must wait for the customer's ready time.

Sets

$C\{1,2,\dots,n\}$: Customer set

$N\{0,1,2,\dots,n\}$: Node set (0. node represent warehouse.)

$D\{0,1,\dots,V\}$: Days set

Decision Variables

t_i : arrival time to customer i

w_i : waiting time at customer i

s_i : service time of customer i

x_{ijd} : a binary variable whether the vehicle travels from point i to j on day d (1 if the vehicle is going from i customer to j customer on day d, 0 otherwise)

Parameters

d_{ij} : distance between customer i and customer j

Q : vehicle capacity

e_j : start time of time window of customer j

l_j : expiry time of time window of customer j

l_0 : end time of the time window of the warehouse (maximum travel time of the vehicle)

m_i : demand quantity of customer i

$$\sum_{i=0}^n \sum_{j=0}^n \sum_{d=0}^n d_{ij} * x_{ijd} \quad (1)$$

$$\sum_{d=1}^V \sum_{i=0, i \neq j}^n x_{ijd} = 1 \quad (2)$$

$$\sum_{d=1}^V \sum_{i=1}^n x_{i0d} = \sum_{d=1}^V \sum_{j=1}^n x_{0jd} = V \quad (3)$$

$$\sum_{i=1}^n \sum_{j=0, j \neq i}^n m_i * x_{ijd} \leq Q, \forall d \in V \quad (4)$$

$$\sum_{i=0}^n \sum_{j=0, j \neq i}^n x_{ijd} (d_i + s_i + w_i) \leq l_o, \forall d \in V \quad (5)$$

$$w_i = \max\{e_i - t_p, 0\} \quad (6)$$

$$e_j \leq t_j + w_j \leq l_j, \forall d \in V \quad (7)$$

$$x_{ijd} \in \{0, 1\} \quad (8)$$

$$t_j \geq 0 \quad (9)$$

$$w_i \geq 0 \quad (10)$$

Equation 1 is the objective function and aims to minimize the sum of distances. Equation 2 ensures that vehicle is assigned to each customer. Equation 3 allows that the number of vehicles leaving and entering the warehouse is equal to one for each day. In Equation 4, the total of customer demands on the route assigned for each day should not exceed the vehicle capacity. Equation 5 states that all time spent along the route should not exceed the vehicle's maximum travel time. Equation 6 shows the waiting time calculation if the vehicle arrives before the customer is ready. Equation 7 is the time window constraint and Equation 8-10 specifies the value sets for variables.

3.3. Tabu Search Algorithm

Tabu search (TA) is a metaheuristic algorithm based on problem solving and includes the concept of 'memory'. The basic approach is to ban or punish repetition in the next cycle to prevent the step leading to the final solution from making circular motions. Thus, the Tabu Search algorithm guides the

regional-heuristic research to search for solutions that are beyond the regional optimum by examining new solutions. Tabu search algorithms can be used for the problems of scheduling, investment planning, communication, design, routing, etc. (Glover and Laguna, 1998). The basic principle of TA is to create a tabu list to prevent a previous move from reversing rather than repeating it. Eq. (3.1) can be used to explain the TA:

$$\text{Min } c(x), x \in X \quad (3.1)$$

Here each x element represents a motion, and the entire set of motions is denoted by X . The x vectors are used as the TA memory structure. Thus, depending on the memory value kept in the vector, some movements will be considered as taboo in the search for solutions, and some will focus more on others. Each movement in the vector X represents the choice of a neighbor of the current solution.

Some concepts and steps of TA method are explained in detail below (Jayaswal, 2000):

Creating a Solution: Initial solution pairs were determined randomly.

Movement Mechanism: The changes made in the solution provide a new solution. There is a stopover from the current point to the visit of all points. This event constitutes the neighbors of the current solution.

Search Neighbor: The most crucial point provided by the neighbor search concept is that it provides the best motion selection. Neighborhood is chosen by the best value chosen in each iteration. Besides, let the neighborhood of x to be $N(x)$, it could be expressed as $N^*(x) \subset N(x)$.

Aspiration criteria: This criterion is the key factor in defining the degree of applicability of the method and allows for the action that provides better results (Salhi, 2002). If $f(x_{\text{tabu}}) < f(x_{\text{best}})$, then current movement will be invalid.

Memory: It memorizes prohibited actions and adds them to the tabu list.

Taboo List: Tabu list adds solutions that have been visited before to avoid duplication. The size of the tabu list is very important.

Stopping Criteria: Reaching maximum iteration number or reaching the desired solution, etc. situations provide stop condition for tabu search.

The steps of the TA algorithm are simply shown below.

Step 1: Creating the initial solution
 Step 2: Creating a candidate solution list
 Step 3: Evaluate the solutions
 Step 4: Choose the best solution
 Step 5: Stop criteria satisfied? If non-satisfied, go to Step 2
 Step 6: Save the last solution

4. Computational Results

4.1. Data Sets

The data set is obtained from <http://vrp.galgos.inf.puc-rio.br/index.php/en/> called as Benchmark: Set A. The pseudo code of the proposed TA model was given in Javascript in detail. The processor model of the computer used is Intel® XEON® CPU E5-2660 V2. It also has 2.2 GHz processor base

Algorithm // pseudocode

```
function TabuAlgo()
{
  try
  {
    UnVisitedNodeCount=GetCountUnvisitedNode();/*Returning Unvisited node counts. Its 38 for beginning*/
    while(ToNumber(UnVisitedNodeCount)!=0)/*This loop will continue until Count 0*/
    {
      for(i=1;i<39;i++)
      {
        ChechVisitedStation(InputsChechVisit, OutputsVisit); /*Checking node visited status. If node was
        viseted return Y*/
        VisitedFlag=OutputsVisit.GetProperty(«VisitFlag»);
        if(VisitedFlag==N){
          InputsCoorditanes.SetProperty(«Node», i);
          GetNodeCoorditanes(InputsCoorditanes, OutputsCoorditanes);/*Returning Coordinate Values for
          nodes*/

          GetDistance(InputsDistance, OutputsDistance);/*Calculated distance*/
          DistanceValue = OutputsDistance.GetProperty(«DistanceValue»);
          if(ToNumber(BestDistance)>ToNumber(DistanceValue)){
            BestDistance=DistanceValue;
            BestSolution=i;
            Node=i;/*Best Solution*/
            RootX1=x2;
            RootY1=y2;
          }
        }
      }
      GetTimeAndOrderFromNode(InputsValidation,OutputsValidation);/*Calculating Time and Order
      values for this root node*/
      TotalTime=ToNumber(Time)+ToNumber(TotalTime);
      TotalOrder=ToNumber(Order)+ToNumber(TotalOrder);
      if(ToNumber(ValidTime)>ToNumber(TotalTime) && ToNumber(ValidOrder)>ToNumber(TotalOrder))
      {
        AddBestPathNode(Node, OutputsBestPath);/*Added Path and flagged as visited*/
        Station=Station+1; /*Rised Station values for same day*/
        x1=RootX1;/*New root Node X1 updated*/
        y1=RootY1;/*New root Node Y1 updated*/
        DistanceValue=0;
        BestDistance=0;
        InputsList.SetProperty(«Node», Node);
        UpdateVisitedList(InputsList, OutputsList); /*Update visited Node list*/
        UnVisitedNodeCount=ToNumber(UnVisitedNodeCount)-1;
      }
      else {/*Return Store and begining values also Day Values rised*/
        x1=39;
        y1=19;
        TotalTime=0;
        Time=0;
        TotalOrder=0;
        Order=0;
        Station=1;
        Day=Day+1;
      }
    }
  }
}
```

Figure 1. Pseudocode of the proposed algorithm

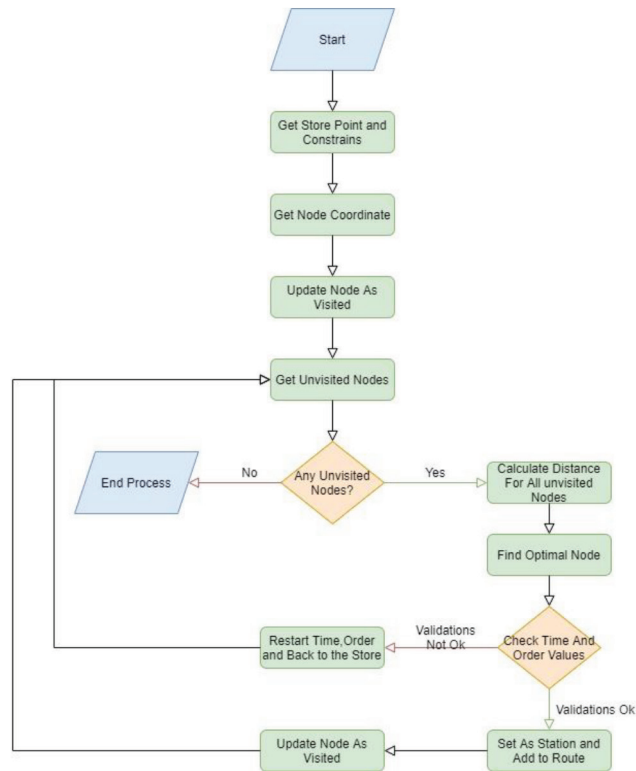


Figure 2. Process Flow Chart

frequency. The pseudo code of the TA which is employed to solve the routing problem is given in Figure 1. At the same, the progress of the algorithm is given as flow chart in Figure 2. Some performance criteria are used to analyze mathematical models. In this study, average solution time was chosen as a performance criterion.

Customer coordinates and demands are presented in Table 1 and Table 2.

Table 1. Coordinates of Customer

	X	Y		X	Y
1	39	19	21	55	43
2	79	19	22	83	29
3	41	79	23	93	49
4	25	31	24	87	23
5	63	93	25	31	23
6	33	5	26	19	97

	X	Y		X	Y
7	69	17	27	41	9
8	57	73	28	83	61
9	53	75	29	9	7
10	1	1	30	13	13
11	79	73	31	43	37
12	59	5	32	13	61
13	1	37	33	71	51
14	41	31	34	45	93
15	23	73	35	93	55
16	37	27	36	5	97
17	85	93	37	81	11
18	93	13	38	7	53
19	85	45	39	7	41
20	49	91			

Table 2. Customer Demand

Warehouse	0	20	7
1	18	21	11
2	16	22	11
3	22	23	1
4	24	24	22
5	3	25	16
6	19	26	15
7	6	27	7
8	6	28	5
9	6	29	22
10	12	30	9
11	18	31	10
12	16	32	11
13	72	33	9
14	7	34	3
15	16	35	7
16	23	36	15
17	4	37	10
18	22	38	2
19	23		

The calculation of the cost was made as the sum of the distances on the route assigned to the vehicle. Euclidean distance is used in calculating the distance as shown in Eq.11.

$$d(A, B) = \sqrt{\quad} \quad (11)$$

The results of the algorithm has summarized in the tables below. Table 3 shows on which day the vehicle will go to which points. For example, on day 1, the vehicle for customer number 16 should leave from the warehouse. The demand for this visit is seen as 16 (See Table 4). Again, on the 2nd day, a demand for 22 units must be delivered to the customer number 25 from the warehouse. The costs for all warehouses are shown in Table 5.

Table 3. Customers to be visited day by day

Assignments	1st day	2nd day	3rd day	4th day	5th day	6th day
I	16	25	27	21	2	20
II	14	4	6	33	37	34
III	31	39	30	19	18	5
IV		13	29	23	24	26
V		38	10	35	22	36
VI		32	12	28	8	
VII		15	7	11	9	

Table 4. Demands of the customers

Demand Section (unit)	1st day	2nd day	3rd day	4th day	5th day	6th day
I	16	22	16	7	18	23
II	72	22	3	11	15	9
III	9	2	22	22	4	24
IV		16	5	11	1	16
V		10	6	3	11	7
VI		10	18	7	6	
VII		7	19	12	6	
VIII				23	16	
Capacity Utilization (%)	97	89	89	96	77	79

Table 5. Total cost by day (distances)

Warehouse	1st day	2nd day	3rd day	4th day	5th day	6th day
I	8,246	8,944	10,198	28,844	40	72,691
II	5,656	10	8,944	17,888	8,246	4,4721
III	6,324	20,591	21,540	15,231	12,165	18
IV		7,211	7,2111	8,944	11,661	44,181
V		17,088	10	6	7,211	14
VI		10	58,137	11,661	51,107	
VII		15,620	15,620	12,649	4,472	
VIII				20,880	12,649	
Total Cost	20,22762	89,45514	131,6523	122,1004	147,5137	153,3447

The assigned route for first day is visualized in Figure 3. As explained in the definition of the problem, each route starts with the warehouse and ends with the warehouse. Considering that the maximum capacity of the vehicles is 100 units, it can be said that the vehicle capacity is 97% on the 1st day. The total cost covering all days was found as 664.29 units of road, end of the analysis for the vehicle routing problem. The total solution time of the problem was calculated as 57 seconds.

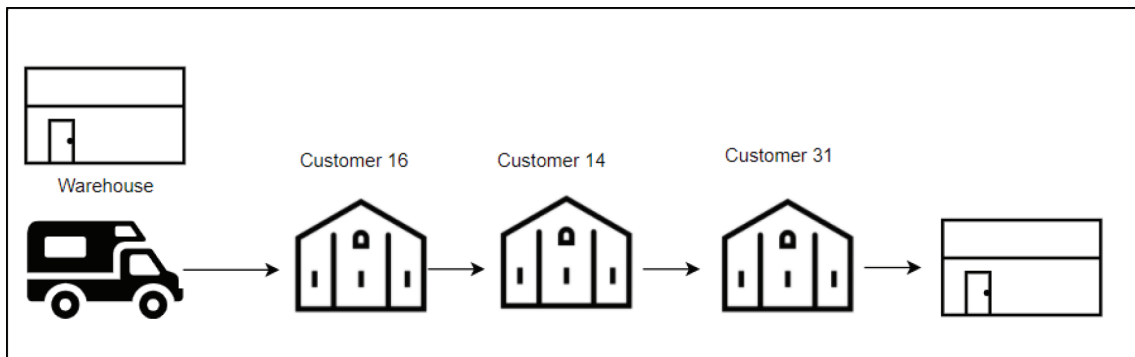


Figure 3. Assigned routes on day 1

5. Results and Discussions

This study proposed a tabu search heuristic for the VRPTWs. The heuristic minimizes total distance traveled using tabu search algorithm. The application of tabu search algorithm, one of the metaheuristic methods frequently encountered in the international literature, to VRPTWs has not been adequately studied in the literature. In order to eliminate this deficiency, information was given about the time windowed vehicle routing problem in the study. In addition, the algorithm obtained the best known solution on every instance of a set of 38 instances of the VRPTWs.

In future work, we plan to solve our tabu search algorithm to the VRPTW problem with Python. Additionally, we programme to compare our method with other metaheuristic algorithms such as genetic

algorithm. This study is assumed that the results of this study will provide various benefits to the practice and the literature.

6. References

- Belfiore, P. Tsugunobu, H. and Yoshizaki, Y. (2008). *Scatter search for Vehicle Routing Problem with Time Windows and Split Deliveries*. *Vehicle Routing Problem*, Book edited by: Tonci Caric and Hrvoje Gold, ISBN 978-953-7619-09-1, pp. 142.
- Bramel, J. and Simchi-Levi, D. (1997). *The Logic of Logistics: Theory, Algorithms and Applications for Logistics Management*.
- Braysy, O. and Gendreau, M. (2002). Tabu Search Heuristics for The Vehicle Routing Problem with Time Windows, *Sociedad de Estadística e Investigación Operativa*, 10 (2), 211-237.
- Cordeau, J.F., Laporte, G., Savelsbergh, M.W.P. and Vigo, D. (2007). Vehicle Routing Transportation, *Handbooks in Operations Research and Management Science*, 14, 367–428.
- Crainic, T.G. and Laporte, G. (1998). *Fleet Management and Logistics*, Springer.
- Glover, F., Laguna, M. (1998). Tabu Search. In: Du, DZ., Pardalos, P.M. (eds) *Handbook of Combinatorial Optimization*. Springer, Boston, MA. https://doi.org/10.1007/978-1-4613-0303-9_33
- Gmira, M., Gendreau, M., Lodi, A. and Potvin, J.Y. (2021). Tabu Search for The Time Dependent Vehicle Routing Problem with Time Windows on A Road Network, *European Journal of Operational Research*, 1(1), 129-140.
- Ho, S. and Haugland, D. (2004). A Tabu Search Heuristic for The Vehicle Routing p-Problem with Time Windows and Split Deliveries, *Computers & Operations Research*, 32, 1947–1964.
- Jayaswal, S. (2000). A Comparative Study of Tabu Search and Simulated Annealing for Travelling Salesman Problem.
- Moccia, L., Cordeau, J.F. and Laporte, G. (2012). An Incremental Tabu Search Heuristic for The Generalized Vehicle Routing Problem with Time Windows, *Journal of the Operational Research Society*, 63(2), 232-244.
- Nguyen, P.K., Crainic, T.G. and Toulouse, M. (2013). Tabu Search for The Time Multi-zone Multi-trip Vehicle Routing Problem with Time Windows, *European Journal of Operational Research*, 231, 43-56.
- Salhi, S. (2002). Defining Tabu List Size and Aspiration Criterion Within Tabu Search Methods, *Computers & Operations Research*, 29(1), 67-68.