Route Determination for Capacitated Vehicle Routing Problem with Two Different Hybrid Heuristic Algorithm

Erhan Baran*

*Aircraft Technology Programme, TAI-Kazan Vocational School, Gazi University, Ankara, 06980

(erhanbaran@gazi.edu.tr)

‡ Erhan Baran, Aircraft Technology Programme, TAI-Kazan Vocational School, Gazi University, Ankara, Tel: +90 312 800 0614,

Fax: +90 312 800 0618, erhanbaran@gazi.edu.tr

Received:11.05.2018 Accepted: 30.06.2018

Abstract: In today's competitive environment, time is very important. Companies that use time effectively always outperform others. Each area has specific methods to use the time effectively. In delivering the goods, it is possible to use the time effectively by carrying out the shipment with appropriate route and vehicle. Proper routing is the most important step of a consignment. In this study, routes were determined for the capacity vehicle locating problems. The tour developers in the literature are compared to each other by making applications for the main ones of the heuristic methods. Comparisons of two different hybrid solution methods have been made. First, the initial solution was developed with the KTA Algorithm, a new algorithm developed in recent years for the determination of routes, and then the route was developed with Van Breedam heuristic methods. In the other hybrid solution, an initial solution was first created with the saving algorithm and a solution was developed using the Kinderwater-Savelsbergh method. Finally, comparison of these two hybrid methods has been done to determine which is more effective in this type of problem.

Keywords Capacitated Vehicle Routing, Kinderwater-Savelsbergh Heuristic, KTA Algorithm, Van Breedam Heuristic, Saving Algorithm.

1. Introduction

Distribution systems have become increasingly complex due to the time-consuming demands of customers and the variable product characteristics. For this reason, the companies that are working in this area have to make their delivery plans more carefully. It is aimed to supply the demands with minimum spending. Car locating problems are a problem of a distribution system. Problems dealing with the distribution of goods between the warehouse and the customers are referred to as vehicle routing problem [1]. Examples are material distribution organization, money transfers, school services, distribution of fuel shipments, and so on. In this study, the problem of capitulant tooling has been investigated. These problems can be described as the problem of shipment with vehicles having the same capacity as the nodes with deterministic demands, with a central depot coordinate set. The solution space consists of unexposed route clusters providing the least cost and constraints for each vehicle. In this study, unlike the studies in the literature, a comparison of two different hybrid solution methods which have not been used previously has been made. First of all, the hybrid approach of Van Breedam Sezgiselin was done with the KTA algorithm from the heuristic methods that develop the tour. Later, a hybrid study of Saving Algorithm and Kinderwater and Savelsbergh heuristic was made and these two hybrids are compared to each other. In the next section, a review of the literature is given for the problem of capitol tooling and the heuristic methods used. Then the solution methods are explained and an application is made about this used method.

2. Literature Review

There are many studies in the literature in this regard. In this section, previous studies on vehicle routing problems in the literature have been examined. Of them; In the study of Wei et al. [2], the capacity-constrained vehicle routing problem, well-known in the literature, has been studied with two-dimensional loading constraints. A minimum cost route has been identified that meets customer demands, including two-dimensional and rectangular product sets, beginning and ending at the central depot. In order to control the effort spent on different routes, an annealing simulation method has been developed in which the temperature is raised and cooled repeatedly. A 2-step heuristic was used to obtain a valid initial solution. For the search process, 4 neighborhoods are preferred. An open-space heuristic method has been used to identify appropriate installation models. In addition, the data structure Trie has enabled the process to accelerate by storing the compliance information of the routes and controlling the effort spent in different routes. The proposed algorithm has been tested in common examples of 2D capacity constrained vehicle locating problems. The results show that the approach in the study gives better results than the previous algorithms for 4 different versions of this problem in the literature. In this study, the solving effects of different loading constraints were also compared and interesting results were obtained. In the study of Banos et al. [3], the problem of time-limited capacity-constrained vehicle locus has been investigated. The minimization of the transport distance and the minimization of the workload imbalance between the used tools and their loads are the objective functions. Numerous algorithms have been proposed to solve single-purpose formulations of this problem, including meta-intuitive approaches that provide high-quality solutions at reasonable work times. However, some authors have analyzed the distance traveled as time progresses by adding other purposes in addition to the objective function. The problem with Donati et al. [4] is the problem of time-dependent vehicle locomotion, a fixed-quantity vehicle fleet depending on travel time, traveling from each node, starting from the source node optimally depending on the time of day when the trip starts. The optimization method involves two hierarchical goals. These; number of keys and total travel minimization. Optimization of the total travel time is a continuous optimization problem that solves the problem by disabling the time domain to the appropriate number of subspace spaces. New time-dependent local search procedures also ensure that the appropriate movements are searched at a fixed time in good conditions. Variable traffic conditions play an important role in this derivation of the classical vehicle routing problem and realistic optimization is achieved. In this study, it has been shown that when time constraints such as time window delivery times are used, solutions that can not be applied with known solutions and the variability of traffic conditions can not be found. Finally, the model has been implemented as a real utility. The model is integrated with a robust shortest path algorithm to calculate the time dependent paths between each client pair of the time dependent model. Molina et al. [5] used the Tchebycheff method for a multi-purpose model with tools in different capacities. The three objective functions are used to reduce carbon dioxide emissions and emissions of air polluting gases such as nitrogen and minimize total internal costs. In this study, we developed a C & W saving-based based algorithm to solve the model when time windows are not considered. Finally, a real application has been implemented to show the effectiveness of the model and the algorithm. In the study of Keskintürk et al. [6], the problem of vehicle locating and the classification of solution methods are discussed. There are various types of vehicle locating problems that are published in the literature and limited in capacity and distance, time windows, discrete delivery, back ordering, periodical delivery and categorized as collect and distribute, and solution methods are reported according to the problem types. The solution methods are definite, classical intuitive and meta intuitive in three groups. When the literature is examined it is thought that it can be a source for researchers working on this issue as it is not compared with a classification study related to the subject. At the end of the study, an application related to solving the problem of limited capacity vehicle routing problem with two intuitive solutions was given and analysis of solution methods was made. In the study of Zhao and Mungwattana et al. [7], the problem of vehicle windowing with time window is addressed. In this type of problem, a two-sided time window is used and the customers with the earliest nearest service start times have been scored. In this work, a hybrid algorithm is developed. Local search algorithm and genetic algorithm have been hybridized. This study has two goals. Minimize the minimum number of vehicles required for these and secondly reduce the total travel time the least. Karagül et al. [8] proposed an algorithm based on Newton's law of attraction. The genetic algorithm is used to obtain the initial population. The proposed algorithm produces initial solutions for vehicle locating problems. In the study of Hosseinabadi et al. [9], the speed of the proposed algorithm and the addition of masses interacting with each other based on the Newtonian gravitational motion laws. Based on the concepts of random search agents, two of the 4 parameters of the physics gravitational force are used. In the study of Ke et al. [10], a relatively new variation of the problem of cumulative-capacity vehicle locating, vehicle locus, has been studied that aims to minimize the total time of arrival of the customer. A two-stage meta-heuristic is preferred. Local search operators are used. Jin et al. [11] used a parallel tabu search algorithm. A cooperative meta-heuristic method is used. In the study of Junqueira et al. [12], a 3-dimensional capacity-constrained vehicle locus problem was investigated. A heuristic algorithm for solving is preferred. In Letchford et al. [13], a powerful multi-stream formulation was developed for the solution. An integer mathematical model is used. In Akpinar [14], a new hybrid meta-heuristic algorithm has been developed. The ant colony algorithm is solved by the fusion of the neighboring search algorithm. In the study of Exposito-Izqueirdo et al. [15], a mathematical model was developed for minimizing the transport costs of the vehicles on the road. In order to be able to work with vehicle locating problems, it is necessary to have an effective initial solution. The intent of the tour making instincts is to complete the tour by starting from a starting point and adding a new point to the lower tour each time until all the points belonging to the probing are included in the tour. These intruders stop searching for a solution when the tour is completed. There are many methods in the literature to create initial solutions. In the study of Keskintürk et al. [16], a study was made on the effects of the initial solutions on the 2-opt algorithm. In this study; the nearest neighbor method, the closest insertion method, the cheapest insertion method, the farthest insertion method, and the Clarke and Wright methods are compared with each other. In Eryavuz et al. [17] a comparison is made between the tour developer's intuition and the stochastic saving algorithm. Again, the results obtained by using the savings algorithm and the use of the nearest neighbor algorithm in obtaining the initial solution have been compared in the study of Keskintürk et al. [6]. A modified k-nearest neighbors algorithm in Mohammed et al. [18] was used in vehicle routing. In the study of Huber and Geiger [19], the neighbor search algorithm was used to determine the route. In the study of Laporte et al. [20], classical and modern intuitions have been compared for solving the problem of vehicle locus. In this study, unlike the studies in the literature, a comparison of two different hybrid solution methods which have not been used previously has been made. First of all, the hybrid approach of Van Breedam Sezgiselin was done with the KTA algorithm from the heuristic methods that develop the tour. Later, a hybrid study of Saving Algorithm and Kinderwater-Savelsbergh Sezgiselin was made and these two hybrids were compared to each other.

3. Methods

There are many methods used to solve vehicle routing problems (VRP). These solution methods are classified in Table 1 below.

Table1.Solution Methods for VRP

Exact Solution Methods	**Heuristic Solution Methods**	
	Traditional Heuristic Methods	**Meta-Heuristic Methods**
Branch and Cut Algorithm	Saving Algorithm	Tabu Search
Dynamic Programming	Nearest Neighbor	Annealing Simulation
Branch and Bound Algorithm	Sweeping Algorithm	Genetic Algorithm
Cluster Partition	Van Breedam Heuristic	Ant Colony Algorithm
Cutting Plane	KTA Algorithm	Artificial Bee Algorithm
	Add Cheapest Algorithm	Local Search Algorithm
	Add Nearest Algorithm	Particle Swarm Optimization
	Kinderwater and Savelsbergh	Fire Beetle Algorithm

3.1. KTA Algorithm

Karagül, Tokat and Aydemir found KTA Heuristic in 2016. The name of the algorithm was determined using the initials of the three creators of the algorithm. The basis of the algorithm is Newton's law of mass gravity. In the literature these algorithms are referred to as artificial physical optimization algorithms. The approach in this study is based on two equations expressing two different heuristic inferences.

$$X_i^c = \frac{q_i \cdot d_i}{\sum q_j} \qquad i = 2,\ldots,n \tag{1}$$

In Equation (1), the states between the warehouse and the customers' places are examined and force calculations are made. Where xi represents the mass gravity of the warehouse and i.customer coordinate, qi is the quantity demanded by the customer, (i = 1 store definition), and d is the customer's warehouse distance (1 st row of distance matrix) and n-1 customer number.

$$X_{ij} = \frac{q_i \cdot d_i + q_j \cdot d_j}{(q_i + q_j) \cdot d_{ij}} \qquad i = 2,\ldots,n-1;\ j = i+1,\ldots,n \tag{2}$$

In Equation (2), the relationship between the locations where the customers are located and the strengths of the masses between the warehouse and the customer sites are considered and developed. After finding the centers of gravity, the weight matrix is formed. Table 2 shows how to calculate the weights calculated in the weight matrix.

Table2. Weight Matrix

Demand (qi)		q_1	q_2	q_3	q_4	q_5	q_6
Customer	**Store**	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**
1 Store	-	X_2^c	X_3^c	X_4^c	X_5^c	X_6^c	X_7^c
2 C1		-	X_{23}	X_{24}	X_{25}	X_{26}	X_{27}
3 C2			-	X_{34}	X_{35}	X_{36}	X_{37}
4 C3				-	X_{45}	X_{46}	X_{47}
5 C4					-	X_{56}	X_{57}
6 C5						-	X_{67}
7 C6							-

The solution of the gravitational force matrix solution is shown in Table 3 below.

Table3. Mass Gravity Matrix

Demand (qi)		1	2	3	4	5	6
Customer	**Store**	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**
1 Store	-	X_2^c	X_3^c	X_4^c	X_5^c	X_6^c	X_7^c
2 C1	X_2^c	X_2^c	X_{23}	X_{24}	X_{25}	X_{26}	X_{27}
3 C2	X_3^c	X_{23}	X_3^c	X_{34}	X_{35}	X_{36}	X_{37}
4 C3	X_4^c	X_{24}	X_{34}	X_4^c	X_{45}	X_{46}	X_{47}
5 C4	X_5^c	X_{25}	X_{35}	X_{45}	X_5^c	X_{56}	X_{57}
6 C5	X_6^c	X_{26}	X_{36}	X_{46}	X_{56}	X_6^c	X_{67}
7 C6	X_7^c	X_{27}	X_{37}	X_{47}	X_{57}	X_{67}	X_7^c

The steps of the algorithm are as follows:
Step 1: The weight points of the customers to the store should be written in places called infinity (-) and zero.
Step2: The customer with the least value in the first line is chosen as the starting point; which is why the warehouse and its customers become close to each other as the center of gravity of the customers with the warehouse shrinks. After selecting the relevant value, that column and that line are closed. If the two values are equal to each other, the node closest to the store is selected.
Step 3: Line operations are executed. As the value increases, two customers get closer. Therefore, the maximum value in that line is selected.
Step 4: The remaining lines are scanned in the same way to obtain an effective solution.
Step 5: Thereafter, each line is sorted from small to large and there is a solution to the vehicle locating problem as appropriate as the number of customers.
Step 6: Routes are obtained by looking at vehicle capacities and compared to find the cost of each route.
Step7: Among the solutions that are moving, it is accepted as the solution having the least financial.

3.2. Van Breedam Heuristic

There are 4 operations that are used in this method. These; Chain Crossing (SC), Chain Changer (SE), Chain Displacement (SR) and Chain Linkage (SM) [23].

3.2.1. Chain Crossing (SC)

In this operation, two edges of two different roots are crossed and the rosette is the state between two chains that change [23].

3.2.2. Chain Swapping(SE)

It is the operation that obtains an adjacency solution by exchanging two chains of points between the two routes [23].

3.2.3. Chain Displacement (SR)

The movement of a chain from one turn to another is defined in this way. Symbolically, the notation is (x, 0) or (0, x). The K parameter, which indicates the maximum chain length, limits the maximum number of stations to be replaced [23].

3.2.4. Chain Linking (SM)

This operation is a mixture of chain swapping and chain shifting operations. This operation selects the best from chain exchange or chain displacement operations during application [24].

In order to assess these operations, two local remediation strategies are provided by Van Breedam.
a) Initial improvement (FI): The goal is to implement the first action that improves the function.
b) Best improvement (BI): All possible movements are examined and selected as the best. Van Breedam then specifies a set of parameters that affect the local remediation process.

Step 1: Initial solution (weak, good).
Step 2: For SR, SM, SE movement types (k = 1 or 2), (k) chain length.
Step 3: Selection strategy (FI, BI).
Step 4: Evaluation process for a string length of K> 1 [25].

3.3. Saving Algotihm

Inspired by the work of Dantzig and Ramser by Clarke and Wright (1964). Tour is the most preferred method among founder intuitions. If the two nodes are not connected to each other, the vehicle leaving the depot will go back and forth to each of the nodes. While the points are connected, the vehicle will go first to the first point and then to the second point. Then the warehouse will come back. Thus, according to the difference between the connection and the connection of two points, a saving will occur. The equation (3) is used to find out the saving amount.

$$S_{ij} = C_{i0} + C_{0j} - C_{ij} \quad (3)$$

Savings are calculated for all node pairs and are sorted from large to small. The nodes with high savings are trying to connect to each other and the storage without exceeding the capacity. A new route is opened when the capacity is exceeded, and unconnected edges are added to a new route [26].

3.4. Kinderwater-Savelsbergh Heuristic

In the heuristic method of Kinderwater and Savelsbergh tours are not considered isolated. Routes and customers can be interchanged between different routes [27]. This change is realized by 3 different operations.

3.4.1. Customer Scrolling

It is the transfer of a client on any route to another route.

3.4.2. Customer Crossing

It is an operation that occurs when two different routes are intersected at one point.

3.4.3. Customer Swapping

Customers on two different routes are relocated among themselves.

4. Application

In this study; three different heuristics have been used for a capacitive vehicle routing problem with a customer number of 6 and vehicle capacity (Q) 26. 1 point node depuder. The amount of warehouse demand is 0. 2, 3, 4, 5, 6 and 7 are shown as customer order and C1, C2, C3, C4, C5 and C6. The distance matrix and customer requirements are shown in Table 4 below.

Table4. Distance Matrix and Customer Demands

Demand (qi)	4	7	5	8	10	6
Customer	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**
1 Store	8	8	8	8	8	8
2 C1	-	4	7	10	5	9
3 C2		-	4	7	10	5
4 C3			-	4	7	10
5 C4				-	4	7
6 C5					-	4
7 C6						-

4.1. KTA-Van Breedam Hybrid Heuristic

First, a route has been achieved by applying KTA intuition and Van Breedam intuition together.

4.1.1. Application of KTA Algorithm

The solution weight matrix obtained when calculating the center of gravity is shown in Table 5 below.

Table5. Solution Weight Matrix

Demand (qi)		4	7	5	8	10	6
Customer	**Store**	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**
1 Store	-	0.8	1.4	1	1.6	2	1.2
2 C1	0.8	0.8	2	1.142	0.8	1.6	0.888
3 C2	1.4	2	1.4	2	1.142	0.8	1.6
4 C3	1	1.142	2	1	2	1.142	0.8
5 C4	1.6	0.8	1.142	2	1.6	2	1.142
6 C5	2	1.6	0.8	1.142	2	2	2
7 C6	1.2	0.888	1.6	0.8	1.142	2	1.2

Then the steps of the method are applied. First, the route 1-2-3-4-5-6-7-1 is obtained. Then all the lines from row 1 to row 7 are sorted in ascending order, resulting in roots. The least costly routing is chosen from these routes. Routes obtained by the KTA algorithm are shown in Table 6.

Table6.KTA Algoritm Routes

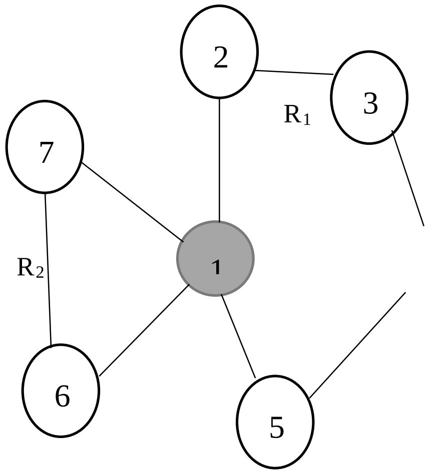
Routes	Costs
R1: 1-2-3-4-5-1 R2: 1-6-7-1	(8+4+4+4+8)=28 (8+4+8)=20 Total=20+28=48
R1: 1-6-5-3-1 R2: 1-7-4-2-1	(8+4+7+8)=27 (8+10+7+8)=33 Total=27+33=60
R1: 1-3-6-4-1 R2: 1-7-2-5-1	(8+10+7+8)=33 (8+9+10+8)=35 Total=33+35=68
R1: 1-2-4-7-3-1 R2: 1-5-6-1	(8+7+10+5+8)=38 (8+4+8)=20 Total=38+20=58
R1: 1-3-5-2-1 R2: 1-6-4-7-1	(8+7+10+8)=33 (8+7+10+8)=33 Total=33+33=66
R1: 1-4-6-5-1 R2: 1-7-3-2-1	(8+7+4+8)=27 (8+5+4+8)=25 Total=27+25=52
R1: 1-7-5-6-1 R2: 1-2-4-3-1	(8+7+4+8)=27 (8+7+4+8)=27 Total=27+27=54
R1: 1-6-3-7-1 R2: 1-5-2-4-1	(8+10+5+8)=31 (8+10+7+8)=33 Total=31+33=64

As can be seen, the least cost route found with the KTA Algorithm is the first route we find. There are 2 vehicles on this route. The route of the first vehicle (R1) is 1-2-3-4-5-1, the route of the second vehicle (R2) is 1-6-7-1.

4.1.2. Application of Van Breedam Heuristic

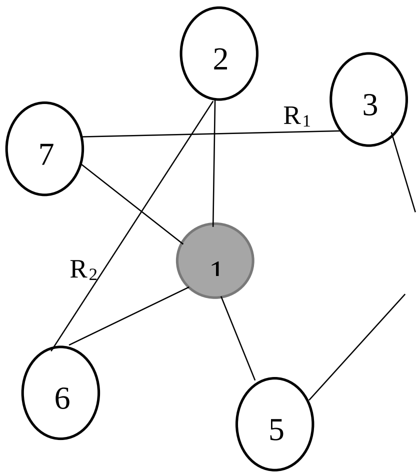
In order to implement Van Breedam Heuristic, an initial solution is needed first. In this study, the solution obtained with KTA Algorithm was used as the initial solution for Van Breedam. The routes taken by the KTA Algorithm are shown in Figure 1 below.

Figure1.Solution of KTA Algorithm



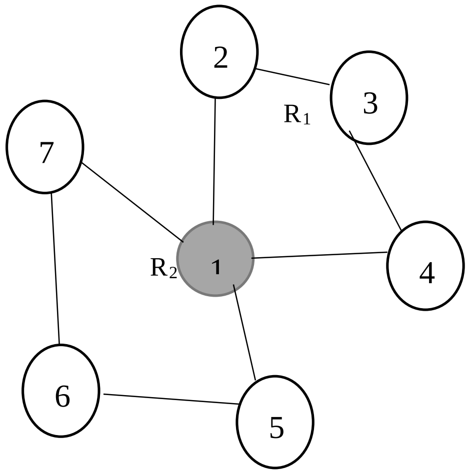
Chain Crossing (SC) operation is performed first. As a result of this operation new rods; R1: 1-7-3-4-5-1, R2: 1-6-2-1. In Figure 2, the route obtained by this operation is given.

Figure2. Solution obtained by chain crossing operation



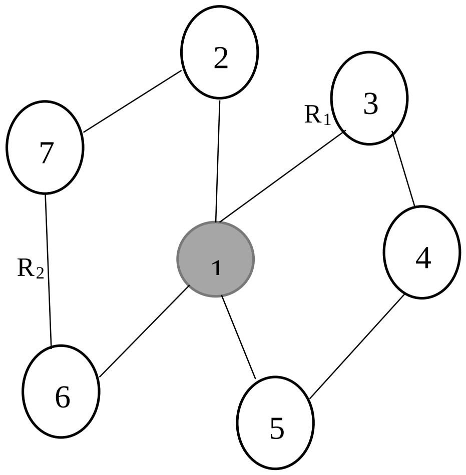
Thereafter, the chain changeover (SE) operation is performed. As a result of this operation new rods; R1: 1-2-3-4-1, R2: 1-5-6-7-1. In Figure 3, the route obtained by this operation is given.

Figure3. Solution obtained by chain exchange operation



Subsequent Chain Displacement (SR) operation is performed. As a result of this operation new rods; R1: 1-3-4-5-1, R2: 1-6-7-2-1. In Figure 4, the route obtained by this operation is given.

Figure4. Solution obtained by chain displacement operation



The chain concatenation operation is a mixture of chain shifting and chain exchange operations. Best choice during application. In order to evaluate these operational movements, Van Breedam offers two local improvements. It will be decided according to these improvements.

4.1.3. KTA-Van Breedam Routes

The roots and chain lengths obtained with the hybrid of KTA-Van Breedam intuitions are given in Table 7 below.

Table7.KTA-Van Breedam Routes

Operation	**Routes**	**Chain Length**
Chain Crossing	R1: 1-7-3-4-5-1 R2: 1-6-2-1	29+21=50
Chain Exchange	R1: 1-2-3-4-1 R2: 1-5-6-7-1	27+24=51
Chain Displacement	R1: 1-3-4-5-1 R2: 1-6-7-2-1	24+29=53
Chain Linking	R1: 1-7-3-4-5-1 R2: 1-6-2-1	29+21=50

The BI strategy was chosen as the election strategy. The best of all possible movements will be selected. The most suitable operator as the chain length is the Chain Crossing Operator. Capacity constraints are not exceeded in these operations.

4.2. Saving Algorithm-Kinderwater and Savelsbergh

In this solution method, the initial solution is determined with the Saving Algorithm. Later, using this initial solution, routes were determined with Van Breedam Intuition.

4.2.1. Application of Saving Algorithm

Vehicle routing according to the saving algorithm will be done for 6 customers with 1 warehouse. Table 8 shows the codes of customers and stores.

Table8. Warehouse and customer codes

Node Name	**Code**
Store	1
Customer 1	2
Customer 2	3
Customer 3	4
Customer 4	5
Customer 5	6
Customer 6	7

In order to implement the savings algorithm, the customer and store codes and the customer's demand information are determined in the first step. Customer requirements are given in Table 9.

Table9. Customers and demand quantities

Node Name	**Code**	**Demand Amount**
Store	1	-
Customer 1	2	4
Customer 2	3	7
Customer 3	4	5
Customer 4	5	8
Customer 5	6	10
Customer 6	7	6

The distance matrix, which specifies the distance between the customer and the warehouse, must be known. In the previous sections, the distance matrix is given in Table 4. The savings amounts (sij) can be calculated by using the distance matrix table. The calculated savings values are shown in Table 10 by creating a Savings Matrix.

Table10.Saving Matrix

	1	2	3	4	5	6	7
1	-	-	-	-	-	-	-
2	-	-	12	9	6	11	7
3	-		-	12	9	6	11
4	-			-	12	9	6
5	-				-	12	9
6	-					-	12
7	-						-

Saving values are sorted from large to small, and after the savings matrix is created, the calculation is first started from the largest savings. Saving Steps are shown in Table 11. The two routes continue to merge the route as long as the requests are met. When the capacity is exceeded, the iteration continues with the next highest saving value.

Table11.Saving Steps

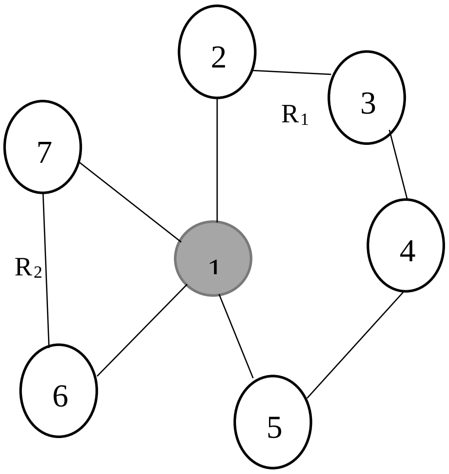
Saving Amount	Position1	Position2	Demand	Decision
12	2	3	11	2-3
12	3	4	17	2-3-4
12	4	5	25	2-3-4-5
12	5	6	35	Capacity Overflow
12	6	7	16	6-7
11	2	6	35	Capacity Overflow
11	3	7	31	Capacity Overflow
9	2	4	17	Same Routes
9	3	5	25	Same Routes
9	4	6	35	Capacity Overflow
9	5	7	31	Capacity Overflow
7	2	7	31	Capacity Overflow
6	2	5	25	Same Routes
6	3	6	35	Capacity Overflow
6	4	7	31	Capacity Overflow

As you can see in Tabloda, the biggest savings amount is 12 and there are 5 different points. As a result of all the iterations, 2 different routes were found. These routes are; 1-2-3-4-5-1, 1-6-7-1. In other words, it was possible to reach all customers with 2 vehicles.

4.2.2. Kinderwater-Savelsbergh Heuristic Application

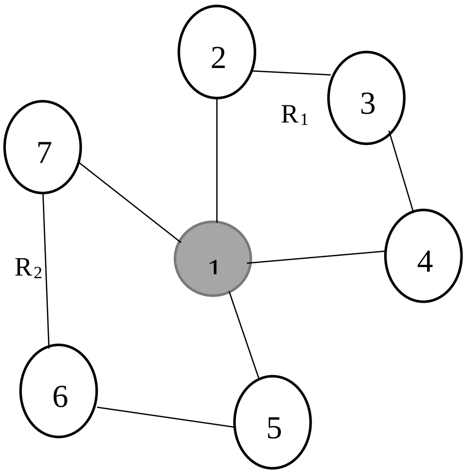
There is a need for an initial solution to solve this method. In this study, the solution obtained by the Saving Algorithm is used as the initial solution. The operation of Kinderwater-Savelsbergh Heuristic has been applied to this initial solution. The routes obtained by the Saving Algorithm are shown in Figure5.

Figure5.Saving Algorithm Solution



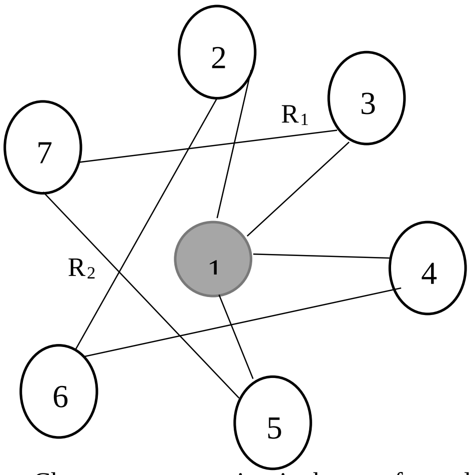
Initially, the Customer Shift operation was performed. The roots of this operation are shown in Fig. As a result of this operation new rods; R1: 1-2-3-4-1, R2: 1-5-6-7-1.

Figure6.Customer Shift Solution



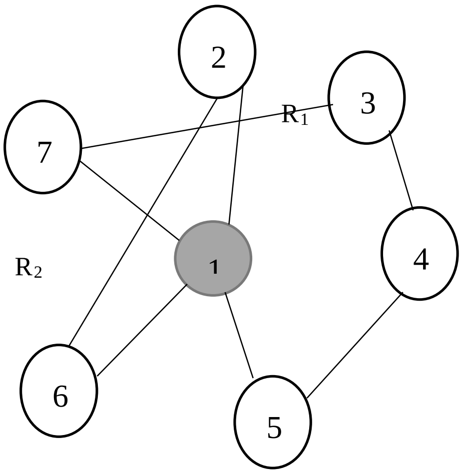
Customer Crossover operation is then performed. As a result of this operation new rods; R1: 1-3-7-5-1, R2: 1-2-6-4-1. In figure 7, the route obtained by this operation is given.

Figure7. Customer cross-over solution



Customer Changeover operation is then performed. As a result of this operation new rods; R1: 1-7-3-4-5-1, R2: 1-6-2-1. The route obtained by this operation is given in Figure8.

Figure8. Customer exchange solution



4.2.3. Saaving Algorithm-Kinderwater Savelsbergh Routes

Routes obtained after the operations performed are given in Table 12.

Table12.Saving Algoritm-Kinderwater and Savelsbergh Routes

Operation	**Routes**	**Chain Length**
Customer Scroll	R1: 1-2-3-4-1 R2: 1-5-6-7-1	24+24=48
Customer Cross-Over	R1: 1-3-7-5-1 R2: 1-2-6-4-1	28+31=59
Customer Swap	R1: 1-7-3-4-5-1 R2: 1-6-2-1	29+21=50

4.3. Comparison of KTA-Van Breedam and Saving Algorithm-Kinderwater Savelsbergh

In this section, the routes found with KTA-Van Breedam Heuristic and the routes with Saving Algorithm-

Kinderwater and Savelsbergh Heuristics are compared. The routes and chain lengths obtained by these two different hybrid heuristics are shown in Table 13.

Table13. Comparison of Two Hybrid Heuristic

KTA-Van Breedam		Saving Algorithm-Kinderwater and Savelsbergh	
Routes	Chain Length	routes	Chain Length
R1: 1-7-3-4-5-1 R2: 1-6-2-1	50	R1: 1-2-3-4-1 R2: 1-5-6-7-1	48

For both hybrid heuristics, the result from the operator from which the best result is obtained is selected.

5. Conclusion

In this study were obtained from two different routes for hybrid vehicle routing heuristic capacity problems. Saving-Kinderwater and Savelsbergh hybrid heuristics seem to give better results for cost-effective vehicle routing problems when comparing the results obtained with KTA-Van Breedam and Saving-Kinderwater and Savelsbergh heuristic. One of the most important reasons for this is that Kinderwater-Savelsbergh heuristic, which is one of the tour developer's heuristic methods, is a very effective method.

References

[1] Toth, P., Vigo, D., 'The vehicle routing problem', SIAM Monographs on Discrete Mathematics and Applications, SIAM Publishing, Philadelphia, PA, 2001.
[2] Wei, L., Zhang, Z., Zhang, D., Leung, S.C.H., 'A simulated annealing algorithm for the capacitated vehicle routing problem with two dimensional loading constraints', European Journal of Operational Research, 1-17, 2017.
[3] Banos, R., Ortega, J., Gil, C., Marquez, A.L., Toro, F.D., 'A hybrid meta-heuristic for multi-objective vehicle routing problems with time windows', Computers & Industrial Engineering, 65, 286-296, 2013.
[4] Donati, A.V., Montemanni, R., Casagrande, N., Rizzoli, A.E., Gambardella, L.M., 'Time dependent vehicle routing problem with a multi ant colony system', European Journal of Operational Research, 185, 1174-1191, 2008.
[5] Molina, J.C., Eguia, I., Racero, J., Guerrero, F., 'Multi-objectivevehicle routing problem with cost and emission functions', Procedia-Social and Behavioral Sciences, 160, 254-263, 2014.
[6] Keskintürk, T., Topuk, N., Özyeşil, O., 'Araç rotalama problemleri ile çözüm yöntemlerinin sınıflandırılması ve bir uygulama', İşletme Bilimi Dergisi, 3(2), 2015.
[7] Mungwattana, A., Manisri, T., Charauenpol, K., Janssens, G.K., 'A solution fort he bi-objective vehicle routing problem with time windows using local search and genetic algorithms', International Journal for Traffic and Transport Engineering, 6(2), 149-158, 2016.
[8] Karagül, K., Tokat, S., Aydemir, E., 'Kapasite kısıtlı araç rotalama problemlerinde başlangıç noktalarının kurulması için yeni bir algoritma', Journal of Engineering Sciences and Design, 4(3), 215-226, 2016.
[9] Hosseinabadi, A.A.R., Rostami, N.S.H., Kardgar, M., Mirkamali, S., Abraham, A., 'A new efficient approach for solving the capacitated vehicle routing problem using the gravitational emulation local search algorithm', Applied Mathematical Modelling, 49, 663-679, 2017.
[10] Ke, L., Feng, Z., 'A two phase metaheuristic for the cumulative capacitated vehicle routing problem', Computers & Operations Research, 40, 633-638, 2013.
[11] Jin, J., Crainic, T.G., Lokketangen, A., 'A cooperative parallel metaheuristic for the capacitated vehicle routing problem', Computers & Industrial Engineering, 44, 33-41, 2014.
[12] Janquiera, L., Morabito R., 'Heuristic algorithms for a three dimensional loading capacitated vehicle routing problem in a carrier', Computers & Industrial Engineering, 88, 110-130, 2015.
[13] Letchford, A.N., Salazar-Gonzalez, J.J., 'Stronger multi-comodity flow formulations of the capacitated vehicle routing problem', European Journal of Operational Research, 244, 730-738, 2015.
[14] Akpinar, S., 'Hybrid large neighbourhood search algorithm for capacitated vehicle routing problem', Expert Systems With Applications, 61, 28-38, 2016.
[15] Exposito-Izquierdo, Rossi, A., Sevaux, M., 'A two-level solution approach to solve the clustered capacitated vehicle routing problem', Computers & Industrial Engineering, 91, 274-280, 2016.
[16] Keskintürk, T., Kiremitçi, B., Kiremitçi, S., '2-opt algoritması ve başlangıç çözümünün algoritma sonuçları üzerindeki etkisi', Endüstri Mühendisliği Dergisi, 27(3), 2-12, 2016.
[17] Eryavuz, M., Gencer, C., 'Araç rotalama problemine ait bir uygulama', Süleyman Demirel Üniversitesi, İktisadi ve İdari Bilimler Fakültesi, 6(1), 139-155, 2001.
[18] Mohammed, M.A., Ghani, M.K.A., Hamed, R.I., Mostafa, S.A., Ibrahim, D.A., Jameel, H.K., Alallah, A.H., 'Solving vehicle routing problem by using improved k-nearest neighborhood algorithm for best solution', Journal of Computational Science, 21, 232-240, 2017.
[19] Huber, S., Geiger, M.J., 'Order matters – a variable neighborhood search fort he swap-body vehicle routing problem', European Journal of Operational Research, 263, 419-445, 2017.
[20] Laporte, G., Gendreau, M., Potvin, J.Y., Semet, F., 'Classical and modern heuristics fort he vehicle routing problem', International Transactions in Operational Research, 7, 285-300, 2000.
[21] Kennedy, J., Eberhart, R.C., 'Particle swarm optimization', Proc. of the IEEE Int. Conference on Neural Networks, 4, 1942-1948, 1995.
[22] Wilke, D.N., 'Analysis of the particle swarm optimization algorithm', Master Thesis, Mechanical and Aeronatutical Engineering, University of Pretoria, 2005.

[23] Yılmaz, Ş., 'Çok depolu araç rotalama probleminin karınca kolonisi optimizasyonu ile modellenmesi ve bir çözüm önerisi', Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2008.
[24] Van Breedam, A., 'Comparing descent heuristics and metaheuristics for the vehicle routing problem', Computers & Operations Research, 28, 289-315.
[25] Laporte, G., Semet, F., 'Classical heuristics for the vehicle routing problem', Les Cahiers du GERAD, G-98-54, Canada, 1999.
[26] Lysgaard, J., 'Clarke & Wright's savings algorithm', Department of Management Science and Logistics, The Aarhus School of Business, Fuglesans Alle 4, DK-8210 Aarhus V, September, 1997.
[27] Gambardella, L.M., Taillard, E.D., Dorigo, M., 'Ant colonies fort he quadratic assignment problem', The Journal of the Operational Research Society, 50(2), 167-176, 1999.