Sigma J Eng & Nat Sci 37 (3), 2019, 827-840

Sigma Journal of Engineering and Natural Sciences
Sigma Mühendislik ve Fen Bilimleri Dergisi

Research Article

SOLUTION APPROACHES FOR MIXED PALLET COLLECTION PROBLEM: A CASE STUDY IN A LOGISTIC COMPANY

Saadettin Erhan KESEN*[1], Muzaffer ALIM[2]

[1]*Konya Technical University, Department of Industrial Engineering, KONYA;* ORCID: 0000-0001-9994-5458
[2]*Technology Faculty, Batman University, BATMAN;* ORCID: 0000-0002-4420-7391

Received: 04.02.2019 Revised: 27.05.2019 Accepted: 05.07.2019

ABSTRACT

In this paper, we study a mixed pallet collection problem in a warehouse of the company operating in fast moving consumer goods industry and present a mixed integer programming formulation with the objective function of total travelling distance minimization. The problem studied is shown to be equivalent to the well-known vehicle routing problem. Since the problem belongs to the class of NP-hard problems, introduced mathematical formulation cannot provide optimal solution in an acceptable amount of time. We, therefore, develop an algorithm based on Simulated Annealing (SA) meta-heuristic approach to find near-optimal solution in a quite shorter computational time. Routes are constructed using Clarke&Wright saving algorithm and then these routes are perturbed whereby three neighborhood operators, namely swap, insert, swap-range are utilized to further improve the quality of the solution. Experimental results based on a real case instance demonstrates that SA algorithm is capable of providing solution more quickly than that of CPLEX solver but the quality of the solution found by SA is 7% worse than that of CPLEX.

Keywords: Vehicle routing problem, Clarke and Wright saving algorithm, simulated annealing.

1. INTRODUCTION

Organizations need to deliver their products and services to the customers in the fastest way in order to be ahead of their competitors under the challenging competitive environment of global market conditions. Nowadays, in the scope of globalization, companies are able to sell their products all over the world and they reach wider distribution regions thanks to the effective distribution channels. At this point, the importance of the concepts of supply chain and logistics has emerged. Logistics is a process consisting of value-added transactions such as transportation, storage and packaging of goods, services and related information between the point of production and consumption points in order to meet the customer requirements. Increasing the customer satisfaction is a must for businesses to become a market leader and this can be achieved by completely eliminating delays with a well-organized logistic plan. While efforts are being made for a better customer satisfaction, it is clear that there should be some methods in use in order to reduce the costs as much as possible.

* Corresponding Author: e-mail: sekesen@ktun.edu.tr, tel: (332) 223 87 06

Storage is one of the leading issues and its importance is getting increased under the content of modern logistic management system. The importance of storage is coming due to its impact on continuity in production for companies and rapid product delivery based on the market dema nds. In this study, we investigate a logistic company which stores more than 1000 kinds of fast moving consumer goods and deliver them to different customer portfolio. Each product type is stored in different locations in the company. Majority of the customers' orders consist of more than one product type. Therefore, the problem of Mixed Pallet Collection Problem (MPCP) is arising in this company. If the demands are not collected in the most appropriate way, there will be an increase on labor costs and a reduction on the effective use of collectors, which leads to an increase in the cost which is an important competitive power for firms. The aim of this study is to use the shortest route to collect customer orders without exceeeding the vehicle capacity. In this respect, MPCP is similar to Vehicle Routing Problem (VRP).

VRP belongs to the class of NP-Hard problems. As the number of customers is increasing in the problem, the difficulty of the problem increases exponentially. In the literature, there are many methods developed for the solution of VRP. These methods are categorized as exact solution algorithms, heuristic and metaheuristic methods. Although exact solution algorithms find optimal solution for small problems, they require more computational time for larger problem sizes. Heuristic methods which can find a solution near to optimal in a limited time are needed as VRP is a difficult problem and encountered frequently in real life. Heuristic methods can produce good quality solutions close to the optimal for large problem sizes with less computational effort and time. The heuristic methods are the most effective algorithms for VRP. Meta-heuristics have been used more in the literature in recent years and they yield successful results. These techniques provide good solutions for combinatorial problems and one or more solutions are usually considered and iteratively produce better solutions from these solutions.

In this study, a method is developed to solve VRP by using the heuristic and metaheuristic methods together. The SA algorithm is coded in the C# programming language and the solution is compared with the solution provided by CPLEX solver.

The remaineder of the paper is organized as follows: Section 2 provides a literature review. Subsequently, problem definition of VRP is presented is Section 3. Section 4 is devoted to providing mathematical formulation. The general description of SA algorithm is described in Section 5. In Section 6, the results of the model based on a real case are shown and discussed.

2. LITERATURE SURVEY

In this section, the studies related to the VRP in the literature are presented. G. Dantzig ve J.Ramser (1959) is the first study in literature for VRP and considered a large scale of Travelling Salesman Problem (TSP) [2]. Next, the mathematical model and solution approach for VRP are presented by considering the vehicle capacity to the TSP. Clarke and Wright (1964) is the first study considering the multiple vehicles [3]. The first article which has a 'Vehicle Routing' in the title is published by Golden, Magnanti, Nguyan (1972) [4]. M. Solomon (1983) has added time window constraints to the Classical Vehicle Routing Problem [5]. The first study on Simultaneous Pick up-Delivery VRP (SPDVRP) is the Min (1989) which is a solution of a real case problem faced by a public library and consists of one depot, two vehicles and 22 customers [6]. Dethloff (2001) describes the relationship between reverse logistic and SPDVRP in environmental protection [7]. Split Demand Simultaneous Pick up and Delivery Vehicle Routing Problem (SDSPDVRP) allows more than one visit to a node and the customers demand can be greater than vehicle capacity. The first study on SDSPDVRP is carried out by Ayşe Bayrak and Bahar Özyörük [8]. Green Vehichle Routing

Problem (GVRP) as a new type of VRP consider the environmental factors such as fuel consumption, gas emission etc. as different to the traditional approach [9].

The first Stochastic Vehicle Routing Problem (SVRP) is a simulation model studied by Cook and Russell (1978) [10]. Although there have been a number of papers in the 1980s about SVRP, the studies in this area have increased in the 1990s. More powerful computers and enhanced coding capabilities have helped researchers to easily model and analyze SVRP.

The first dynamic VRP studied by Powell (1986) [11]. Dynamic vehicle routing was started to be seen more common in the literature in the second half of the 1990s by the development of advanced computational capability, vehicle tracking and data storage in 1990s.

In 1997, Brandão ve Mercer used the Tabu Search algorithm for Multivariate Vehicle Routing and Scheduling problem which is a different type of VRP in which vehicles can serve more than once in a day [12].

For the solution of VRP, two different solution methods have been developed as exact and heuristic methods. The direct tree search by Christofides (1985), integer linear programming by Laporte ve Nobert (1987) and dynamic programming by Laporte (1992) are such methods used to solve VRP [13,14,15]. Furthermore, the method of saving proposed by Clarke ve Wright (1964, sweeping method mentioned by Foster ve Ryan (1976) and developed by Gillet and Miller (1974), a two-stage method developed by Christofides et al. (1979) and petal method suggested by Renaud et al. (1996) are the classical heuristic methods for VRP [16,17,18,19].

In recent years, although the exact methods have been proposed for VRP, they are limited for providing a good solution of problems with up to approximately 100 customers. Since the classical heuristic methods do not provide quality solutions, metaheuristic methods are needed. Metaheuristic methods provide better solutions compared to the classical heuristics but they require more computational time [20]. Artificial Neural Network by Hopfield ve Tank (1985), Tabu Search by Glover (1986), TB by Kirkpatrik vd.(1983), Genetic Algoritm by Holland (1975) and Ant Colony by Dorigo vd. (1991) are the main metaheuristic methods [21,22,23,24,25].

C. Wang et al. (2015) propose the SA algorithm for the VRP with simultaneous delivery and time windows [26]. Gendreau et al. (2006) and Bortfeldt(2012) use the Tabu Search algorithm to solve capacity restricted VRP and 3D loading problems [27,28]. Mester and Braysy (2007) have developed a two-stage method consists of local search and genetic algorithm for the solution of capacitated VRP [29]. E. Aziz (2010) formulates an ant colony algoritm for the solution of periodic vehicle orientation problem with time windows [30]. The solution of dynamic heterogeneous VRP by genetic algorithm and discrete optimization has been studied by E. Edison and T. Shima(2010) [31].

Günther et al. (2015) add time constraint and propose a variable neighbor search algorithm [32]. The studies by Pollaris et al. (2015), Junqueira ve Morabito'nun (2015) argue that restrictions on weight distribution and the conditions for taking and delivery are instructed [33,34]. Table 1 tabulates the historical development of VRP.

Table 1. Historical development of VRP

1950's	VRP is formulated as an integer programming and have been solved for small problems with 10-20 customers.
1960's	Initially, the route bulding heuristic is presented. (Ex. Clarke ve Wright, 1964). For the VRP, 2-opt and 3-opt are applied. Some problems with 30-100 customers are solved.
1970's	A two-phase heuristic array has been proposed (eg, Gillett and Miller, 1974). Computational efficiency has become an issue and has been solved for major problems with 100-1000 customers (örneğin, Golden, Magnanti, and Nguyen, 1977). Some problems with 25-30 customers can be solved by using optimal methods.
1980's	Procedures based on mathematical programming are proposed (eg, Fisher and Jaikumar, 1981). Interactive heuristics are developed (eg, Cullen, Jarvis, and Ratliff, 1981). Some problems with approximately 50 customers have been solved by optimal methods.
1990's	Metaheuristic methods are applied on VRP. Some of the problems with 50-100 customers are solved optimally (Fisher, 1995).
2000's	Metaheuristic methods are integrated into many problems. The usage of Genetic algorithms, artificial intelligence applications and Tabu Search has become more popular and it is getting closer to find optimal solutions.

3. PROBLEM DEFINITION

In this section, the problem and the relevant notation are presented. The aim of the problem is to collect the customers' demands from specific locations by creating a vehicle route and to find the shortest way to do so. Let $G(N, A)$ be a fully connected network for VRP. $N = \{0,1,2, \ldots\ldots, N\}$ represents the nodes and $A = \{(i,j) | i,j \epsilon N$ is for the arc between these nodes. The customers are shown by $N_c = N \backslash \{0\}$ at which "0" is for the depot. Each route should start and end up in the depot and each customer must be visited only once. Each customer in the network has a demand, $d_i : i \in N_c$ and this demand must not exceed the vehicle capacity. There is a cost $c_{ij}: (i,j) \in A$ between each pair of nodes in network. This cost is due to the distance between each pair of nodes. At the same time, there are M vehicles with a capacity limit of Q in the depot and they can be used more than once. General description of the the problem can be seen in Figure 1.

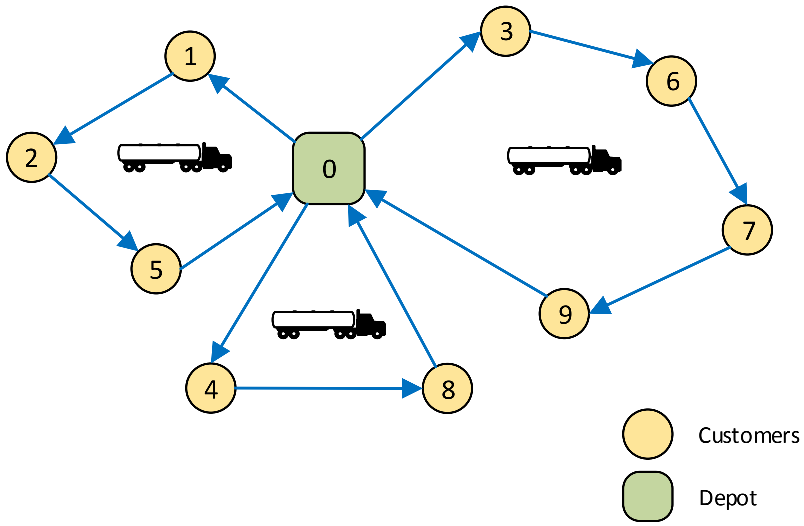

Figure 1. Schematic representation of the vehicle routing problem

3.1. Notations

Indices:

i, j : Nodes ($i, j = 0,1,2, \dots, N$)

Sets and Parameters:

N :Set of nodes

N_c :Set of customers

M :Number of vehicles

Q :Vehicle capacity

d_j :Demand of customer j ($\forall j \in \mathrm{N}_c$)

c_{ij} :The distance between nodes i and j ($\forall i, j \in \mathrm{N}$)

Decision Variables:

$$x_{ij} : \begin{cases} 1, & \text{If the vehicle routes from node } i \text{ to node } j \text{ } (\forall i, j \in N) \\ 0, & otherwise \end{cases}$$

u_i :Auxiliary variable to use for eliminating sub tours and for capacity constraints ($\forall i \in N_c$)

3.2. Mathematical Formulation

This section presents the mathematical formulation of VRP.

$$Min\ Z = \sum_{i,j \in N} c_{ij} x_{ij} \tag{1}$$

st

$$\sum_i x_{ij} = 1 \qquad \forall j \in N_c \tag{2}$$

$$\sum_j x_{ij} = 1 \qquad \forall i \in N_c \tag{3}$$

$$\sum_i x_{i0} = \sum_j x_{0j} = M \tag{4}$$

$$u_i - u_j + Q x_{ij} \le Q - d_j \qquad \forall i, j \in N_c, i \ne j \tag{5}$$

$$d_i \le u_i \le Q \qquad \forall i \in N_c \tag{6}$$

$x_{ij} \in \{0,1\} \qquad \forall i,j \in N_c \qquad (7)$

Eq 1 is the objective function which minimizes the total travelled distance. Eq 2 and Eq 3 shows that there is only one entry and one exit from each node, Eq. 4 satisfies that the number of vehicles entering to a node and leaving from the node is same, Eq. 5 eliminates the sub tours between nodes, Eq. 6 guarantees that the load on the vehicle is higher than the demand and less than the capacity of the vehicle, Eq. 7 is for binary variables.

4. GENERAL OPERATION OF THE ALGORITHM

Our algorithm is heuristic based. First, the initial solution is obtained by Clarke & Wright algorithm. The initial solution is improved by Tabu Search algorithm which is a metaheuristic method. Swap, swap-range and Insert are used as neigbourhood mechanism of SA.

4.1. Clarke and Wright Saving Algorithm

In 1964, Clarke & Wright published an algorithm for the solution of Classical Vehicle Routing Problem. This algorithm is also known as saving algorithm. Clarke-Wright heuristic is constructed based on the saving value generated by returning to the starting point by visiting two points instead of going from the starting point to these points separately. Saving algorithm is a heuristic algorithm so it does not guarantee the optimal solution. However, the method usually offers a relatively good solution. The triangle inequality is used while creating the saving matrix. Figure 2 illustrates the schematic representation of joining routes.

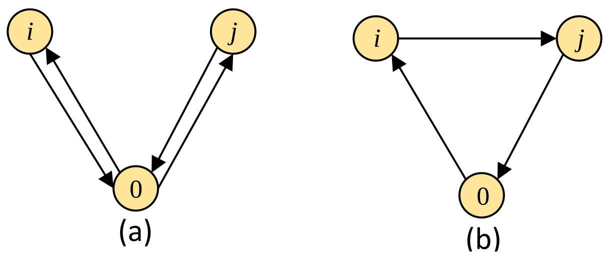

Figure 2. Schematic representation of joining routes

Initially, each customer is served by separate vehicles. For example, the customers i and j in Figure 2.a are visited separately. Alternatively, if the demand of two customers does not exceed the vehicle capacity, they can be visited on the same route. In Figure 2.b, the customers i and j are visited on the same route. Let the transportation cost between two points of i and j be $\boldsymbol{C_{i,j}}$ and the total transportation cost in Figure 2.a as $\boldsymbol{D_a}$:

$$\boldsymbol{D_a = C_{0,i} + C_{i,0} + C_{0,j} + C_{j,0}}$$

D_b represents the total transportation cost in Figure 2.b:

$$\boldsymbol{D_b = C_{0,i} + C_{i,j} + C_{j,0}}$$

The saving by combining two way is $S_{i,j}$:

$$\boldsymbol{S_{i,j} = D_a - D_b = C_{i,0} + C_{0,j} - C_{i,j}}$$

There are two versions of saving algorithm as serial and parallel. In the serial version, one route can be created at a time while multiple routes can be created at one time in the parallel version. At the first step of saving algorithm, savings are calculated for all customer pairs and all customer points are ranked in descending order of savings. Then, one pair at a time is taken into account from the top of the list of ranking pairs. When a pair of i, j is considered, if these two customer points can be connected without deleting a previous established direct connection and the total demand in the resulting route does not exceed the capacity of the vehicle, the routes which visit i and j are connected (so i is visited in the route after j). In a serial version, when a connection is made between a pair, it must be restarted from the top of the list (the combinations that are not feasible until now can become applicable), in a parallel version, it requires only one pass through the list once.

4.2. Simulated Annealing Algorithm

In this study, SA which is one of the stochastic search methods is used in order to use the search space more homogeneously.The SA method is a local search technique which is very popular among artificial intelligence techniques.

SA is a stochastic searh method that provides good solutions for combinatorial optimization problems. The name of SA comes from the similarity of physical annealing process of solids. In other words, it is based on the logic of heating of solids and slowly cooling them. SA algorithm is introduced by, Kirkpatrick,Gerlatt and Vecchi(1983) and Cenny(1985) as independent to each other.

In summary, SA is a stochastic search method which mimics the physical annealing process in which a solid is cooled slowly until its minimum energy state is achieved.

4.2.1. Modeling of simulated annealing

SA starts with the initial solution, i and generates the solution $j$derived from the neighbours of the initial solution. The change in objective function is calculated by $\Delta = f(j) - f(i)$. So it is defined as $\Delta = (neighboring\ solution - existing\ solution)$. If $\Delta = 0$, then the solution is accepted with a certain probability, usually with the probability of $P = exp(-\Delta/T)$. The T value is a control parameter similar to the temperature in annealing. Generally, the value of T is reduced monotonically throughout the algorithm and this is called cooling plan.

The aim of SA algorithm is to find a solution x which optimizes the function of $f(x)$ defined in a subset of all possible solution points, (S). SA algorithm starts with a randomly selected initial solution. Then, a neighbor solution to the existing solution is selected by an appropriate mechanism and the change in $f(x)$ is calculated. If the change is in the intended direction, the neighboring solution is assigned as current solution. If there is no change in the desired direction, the SA algorithm considers this solution as a probability value obtained by "Metropolis Criteria". The acceptance of the solution which creates an inverse direction on objective function with a certain probability allows SA algorithm to avoid local optimals.

4.2.2. The steps of simulated annealing algoritm

A reliable search heuristic is the one with the low dependencies to the initial point. While the solution space is being scanned in the SA, the upward movements which are the movements against the objective value can be controlled by a changing probability based function. SA is an iterative improvement algorithm. The basic steps of a standard SA algorithm are given in Figure 3.

Procedure: Simulated Annealing Algoritm
Input : Clarke & Wright heuristic solution,
M:Number of neighbour solutions
T_b:Initial tempreture
T_f:End temperature
max_it:Maximum number of iteration
r:Tempreature reduction rate
Output: s_{opt}:Optimal solution
f_{opt}:Objective value of the optimal solution
Step 1: Get a initial solution by Clarke & Wright heuristic: $s_0 \in S$ and calculate objective function $f(s_0)$, assign as existing and best solution.
$s \leftarrow s_0$; $f(s) \longleftarrow f(s_0)$;
$s_{iyi} \leftarrow s_0$; $f(s_{iyi}) \longleftarrow f(s_0)$;
Step 2: Choose an initial tempreature: $T_b > 0$
Set the number of neighbouring solution searched at each temperature:M
Set the maximum number of iterations: max_it
Set the temperature reduction rate: r
Reset the temperature change counter $T_d \leftarrow 0$
Step 3:
Step 3.1: Assign the starting temperature as temperature variable. $T_b \leftarrow T_d$
Step 3.2: Reset the actual move and total move counters
$(nrep \leftarrow 0; iterasyon \leftarrow 0)$
Step 3.3: If $T_d <= T_f$ then go to Step 4.
Step 3.4: Randomly generate solution s' ($s' \in N(s)$) which is a neighbor of s, calculate $f(s')$.
$\Delta \leftarrow f(s') - f(s)$.
Step 3.5: If $\Delta <= 0$, assign neighbor solution as current solution ($s \leftarrow s'$),
$nrep \leftarrow nrep + 1$. Go to Step 3.6.
Else (0,1) Produce a random number (u) from the uniform distribution in the range of (0,1).
Step 3.5.1 If $u < exp(-\Delta / T)$, assign neighbor solution as current solution. ($s \leftarrow s'$)
. $nrep \leftarrow nrep + 1$ and go to Step 3.6**.**
Else go to Step 3.7.
Step 3.6: If $f(s') < f(s_{iyi})$ then$f(s_{opt}) \leftarrow f(s')$
Step 3.7: $iteration = iteration + 1$.
If ($nrep > M$) or ($iterasyon > max\_it$) then $T_d = r.T_d$.
go to Step 3.3 .
Else go to Step 3.4.
Step 4: Print the best solution.
Step 5: STOP.

Figure 3. The pseudo-code of simulated annealing algoritm

M: Number of neighbour solutions to be searched at each temperature
T_d: The temperature value at each iteration
The convergence rate of the SA algorithm to the global optimal solution is determined M and $T_d, i = 0,1,2$ .... parameters.

Some parameters need to be determined in order to use the SA algorithm to solve any problem and these are:

(1) Initial Temperature (T_b),
(2) Length of iteration at each temperature,
(3) Cooling function,
(4) Stopping criteria for algorithm

Initial temperature is an input parameter. Temperature is used to control the probability of accepting of the bad solutions. Length of iteration is the number of solutions produced at each temperature. Cooling function determines the temperature at current iteration depending on the temperature of previous iteration. The cooling rate is shown by *r*. In applications, the value of *r* usually is between 0,8 and 0,99. The number of iteration and cooling function with the inital temperature are called cooling schedule. This schedule has a great impact on the quality of solution and convergence rate. If the solution obtained at each temperature change does not change for a number of consecutive temperature changes, then SA is stopped.

4.3. Neighbourhood Structures

The neighbourhood mechanisms are used for the heuristic solutions of NP-hard problems such as vehicle routing, travelling salesman, scheduling etc. For the NP-hard problems, polynomial time algorithm to solve all the samples of that problem class to optimality is not known. At this point, two features are stand out (i) computational time, (ii) quality of solution. The aim is to run the mechanism quickly and obtain near optimal solution. The neighbourhood mechanisms are widely used in computer programming languages due to their simple coding and faster operation of the algorithm. In

Figure 4, there is a sample route for VRP.

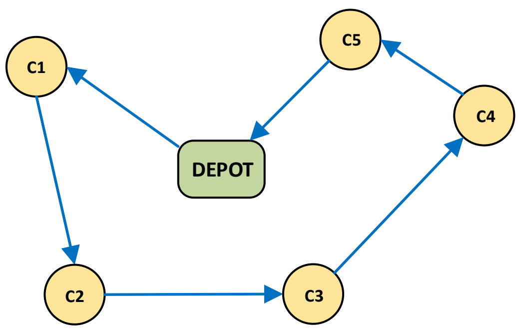

Figure 4. A sample route for ARP

4.3.1. Swap operator

Consider a route, R1 consisting of 5 customers as below. 2 customers are chosen randomly (C5-C4). The customer C5 and C4 are swapped and it creates route R1’ as in Figure 5’

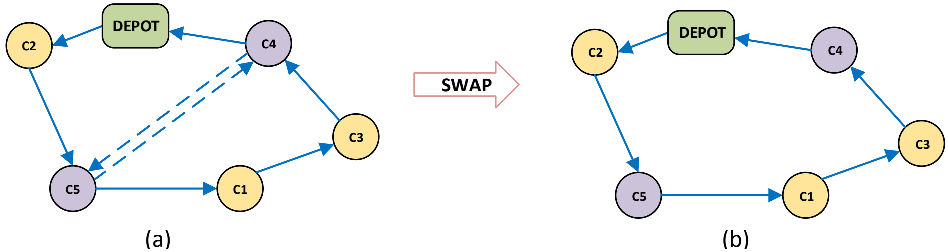

Figure 5. Schematic representation of the swap operator a) R1 route, b) R1' route

Let the distance different between routes R1 and R1' be F_1 and it is calculated as ;

$$F_1 = (d_{C2,C4} + d_{C4,C1} + d_{C3,C5} + d_{C5,depot}) - (d_{C2,C5} + d_{C5,C1} + d_{C3,C4} + d_{C4,depot})$$

4.3.2. Insert operator

Consider the route R1 consisting of 5 customers as above. 2 random customers arechosen randomly (C5-C4). The customer C4 is added to the left side of the customer C5 and the customers on the righ side of the customer C5 move one unit to the right. This creates route R2' and it is illustrated in Figure 6.

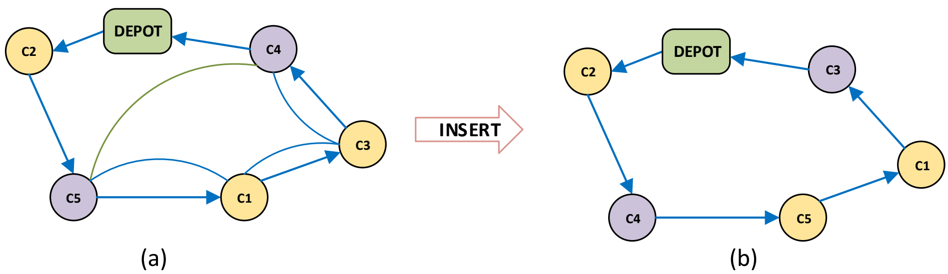

Figure 6. Schematic representation of the ınsert operator a) R1 route, b) R2' route

Let the distance different between routes R1 and R2' be F_2 and it is calculated as;

$$F_2 = \left(d_{C2,C4} + d_{C4,C5} + d_{C3,depot}\right) - (d_{C2,C5} + d_{C3,C4} + d_{C4,depot})$$

4.3.3. Swap-Range operator

Let's refer to the route R1 consisting of 5 customers as above. 2 random customers are chosen randomly (C5-C4). The customers between C5 and C4 (including C5 and C4) are sorted in the reverse order. This creates R3' route as in Figure 7.

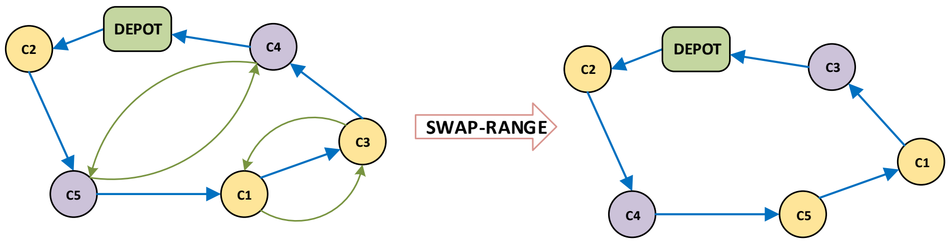

Figure 7. Schematic representation of the swap-range operator a) R1 Route, b) R3' route

Let the distance difference between routes R1 and R3' be F_3 and it is calculated as;

$$F_3 = (d_{C2,C4} + d_{C5,depot}) - (d_{C2,C5} + d_{C4,depot})$$

5. EXPERIMENTAL STUDY

In this section, a real world problem based on a warehouse of a logistic company operating in a fast moving consumer goods sector is addressed. The problem consists of a total of 221 nodes which are 220 customers and 1 depot. The demands of customers are collected monthly and then mixed pallets are collected at the end of each month. We assume that we have the customers' demands at the beginning of a month and the orders received during the month are transferred to the next month. Planning is made at the beginning of a month. As we have the information of the size of all ordered products, we calculate the volume that will cover in the pallet. The aim is to minimize the distance while meeting the customers demand. Vehicle capacity is 1 pallet and vehicles can be used more than once. For the solution of the problem, an initial solution is obtained by Clarke & Wright heuristic. Then the initial solution is improved by SA heuristic. In SA, 3 different neigbourhood mechanisms such as Swap, Swap-Range and Insert are used. The SA algorithm is coded with C# Visual Studio.

The distance matrix for each location pair is calculated in the region in which a depot and 220 locations are located. The mixed integer mathematical model is solved by using CPLEX solver in GAMS interface while limiting the computational time for 2 hours (7200 seconds) and CPLEX has created 103 routes during this time. The upper limit for the total transportation distance of all routes is 18816 while the lower limit is 3575. Based on this, the relative gap is calculated as (upper limit – lower limit)/upper limit=0,81. The higher value of relative gap means higher the difficulty of the problem. As the lower limit obtained by CPLEX does not guarantee a proper solutions, we need to take the upper limit value as a reference point for comparison.

The same dataset is solved by using Simulated Annealing algorithm. The Clarke and Wright algorithm used in SA algorithm for finding initial routes creates 106 different routes and total travelled distance for these routes is calculated as 23335. The routes obtained by Clarke and Wright are improved by SA algorithm which has the parameter values as below.

Initial Temperature: 1000℃
Stopping Temperature: 20℃

The number of neighboring solution to be searched at each temperature: 500

Tempereture reduction rate: 0,97

In the SA algorithm, the routes are improved by using insert, swap-range and swap operators. It can be seen how three different operators can improve the solution within the computational time. 20259 for insert, 22868 for swap-range and 22348 for swap are obtained. The change of the solutions by each temperature change is shown in the convergence graph of Figure 8.

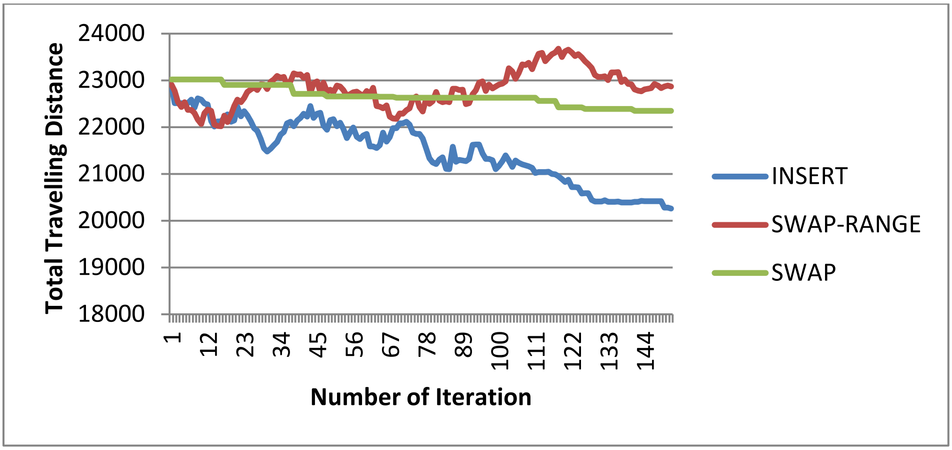

Figure 8. Comparison of the results of neighborhood structures

The SA algorithm obtained the solution in 10 minutes. Compared to the CPLEX, SA algorithm reached to a solution in a very short time and the quality of the solution is %7,6 [(20259-18816)/18816=0,076] which results 7.6% higher transportation cost. As the frequency of mixed pallet collecting process increases, the use of SA algorithm becomes more attractive due to its low computational time.

6. CONCLUDING REMARKS AND FUTURE DIRECTIONS

The companies operating in the logistic sector need to distinguish the areas of production activities and warehouse operations due to high variety and volume of the products. These companies load the customers' orders in pallets to the transportation vehicles in depot area. When preparing the customers orders, we tackle with two basic situations: If the customer's order for any product is in the form of multiples of a pallet, the product is directly loaded to the vehicle. However, nowadays the customers' orders usually consist of many products which do not exceed a pallet. In this case, so called mixed pallet needs to be prepared and another problem arises to decide of which order of the products should be collected to make the pallet ready for shipment. The main criteria while preparing the mixed pallets is to minimize the total transportation distance.

In this study, the mixed pallet collection problem is discussed in the field where a logistic company operates its storage operations and first a mixed integer mathematical model is developed for this problem. Since the problem is in the class of NP-hard problems, the developed model could not provide an optimal solution in polynomial time. Thus, Simulated Annealing method has been developed to obtain good solution for the problem in a quite shorter time. The initial routes in this algorithm are created by Clarke and Wright algorithm and they have been improved by using insert, swap-range and swap neighbour operators. In the experimental studies, it has been shown that the insert operator gives the best results among these three operators. Additionally, the results of the mathematical model coded in CPLEX solver are compared with the results of Simulated Annealing method. The Simulated Annealing algorithm provides solution in a very shorter time compared to CPLEX and solution quality is 7% worse than optimal value. At this point, the decision maker either waits longer to get better solution by mathematical model or accepts a slightly worse result by Simulated Annealing. The use of algorithm may become more attractive as long as the solution for the problem is frequently needed.

The problem handled in the paper can be extended in several points: In future studies, different meta-heuristic algorithms namely, Tabu Search, Genetic Algorithm, Ant Colony Optimization, Variable Neighborhood search can be compared between each other. Furthermore, to minimize the number of pallets used, the problem of bin-packing and mixed pallet collection problem can be jointly considered.

REFERENCES

[1] Şahin Y., Eroğlu A., Kapasite Kısıtlı Araç Rotalama Problemi İçin Meta Sezgisel Yöntemler Bilimsel Yazın Taraması, Süleyman Demirel Üniversitesi İktisadi ve İdari bilimler Fakültesi Dergisi, 19(4), 337-355, 2014.

[2] Dantzig G., Ramser J., The Truck Dispatching Problem, Management Science, 6(1), 80-91, 1959.

[3] Clarke G., Wright J.W., Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. Operations Research, 12(4), 568-581, 1964.

[4] Golden B.L., Magnanti T.L., Nguyen H.Q., Implementing vehicle routing algorithms, Networks, 7(2), 113–148, 1972.

[5] Solomon M., Vehicle routing and scheduling with time window constraints: Models and algorithms, Techinical Report, Northeastern University College of Business Admin, 1983.

[6] Min H., The multiple vehicle routing problem with simultaneous delivery and pickup points, Transportation Research, 23(5), 377–386, 1989.

[7] Dethloff J., Vehicle routing and reverse logistics: The vehicle routing problem with simultaneous delivery and pick-up, OR Spektrum, 23(1), 2001, 79–96, 2001.

[8] Bayrak A., Özyörük B.,, Bölünmüş talepli eş zamanlı topla dağıt araç rotalama problemi için karşılaştırmalı matematiksel modeller, Journal of the Faculty of Engineering and Architecture of Gazi University, 32(2), 469-479, 2017.

[9] Belbağ, S., Yeşil kapasite kısıtlı araç rotalama problemi:Bir literatür taraması, Gazi Üniversitesi İktisadi ve İdari Bilimler Fakültesi,19(1), 345-366, 2017.

[10] Cook T. M., Russell R. A., A simulation and statistical analysis of stochastic vehicle routing with timing constraints, Decision Sciences, 9(4), 673–687, 1978.

[11] Powell W. B., A stochastic model of the dynamic vehicle allocation problem, Transportation Science, 120(2), 117–129, 1986.

[12] Brandao J., Mercer A., A Tabu search algorithm for the multi-trip vehicle routing and scheduling problem, European Journal of Operational Research, 100(1), 180–192, 1997.

[13] Christofides N.,Vehicle scheduling and routing, Presented at the 12th International Symposium on Mathematical Programming, 1985.

[14] Laporte G., Nobert Y., Exact Algorithms for the Vehicle Routing Problem, North-Holland Mathematics Studies, 132, 147-184, 1987.

[15] Laporte G.,The Vehicle Routing Problem: An overview of exact and approximate algorithms, European Journal of Operational Research, 59(3), 345-358, 1992.

[16] Foster B.A., Ryan D.M., "An integer programming approach to the vehicle scheduling problem", Operational Research Quarterly, 27(2), 367-384, 1976.

[17] Gillet B.E., Miller L.R., "A heuristic algorithm for the vehicle dispatch problem," Operations Research, 22(2), 340-349, *1974.*

[18] Christofides N., Mingozzi A., Toth P., The Vehicle Routing Problem In Combinatorial Optimization, Wiley Chichester, 1979.

[19] Renaud J., Boctor F.F., Laporte G., An Improved Petal Heuristic for the Vehicle Routing Problem, Journal of Operational Research Society, 47(2), 329-336 , 1996.

[20] Laporte, G., Ropke, S., & Vidal, T. Heuristics for the vehicle routing problem. Vehicle Routing: Problems, Methods, and Applications, 18, 87, 2014.

[21] Hopfield, J.J., Tank, D.W.,Neural Computation of Decisions in Optimization Problems.

Biological Cybernetics, 52, 141-152,1985.

[22] Glover F., McMillan C., The General Employee Scheduling Problem: An Integration of Management Science and Artificial Intelligence, Computers and Operations Research, 13(5), 563-593, 1986.

[23] Kirkpatrick S.Gelatt Jr. C.D., Vecchi M.P., Optimization by simulated annealing, Science, 220(4598), 671–680, 1983.

[24] Holland J.H., Adaptation in Natural and Artificial Systems, University of Michigan Press, 1975.

[25] Dorigo M., Maniezzo V., Colorni A., Positive Feedback as a Search Strategy, Technical Report, 91-016, 1991.

[26] Wang C., Dong M., Zhao F., Sutherland J.W., A Parallel Simulated Annealing Method For The Vehicle Routing Problem With Simultaneous Pickup–Delivery And Time Windows, Computers And Industrial Engineering, 83,111-122, 2015.

[27] Gendreau M., Guertin F., Potvin J.Y. Seguin R., Neighborhood Search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries, Transport Research Part C: Emerging Techologies, 14(3), 157-174, 2006.

[28] Bortfeldt A., A hybrid algorithm for the capacitated vehicle routing problem with three-dimensional loading constraints, Computers & Operations Research, 39(9), 2248-2257, 2012.

[29] Mester, D., Braysy, O., Active-guided evolution strategies for large-scale capacitated vehicle routing problems, Computers & Operations Research,34 (10), 2964-2975, 2007.

[30] Aziz E., An Algorithm for the Vehicle Problem, International Journal of Advanced Robotic Systems, 7(2), 125-132, 2010.

[31] Edison E., Shima T., Integrated task assignment and path optimization for cooperating uninhabited aerial vehicles using genetic algorithms, Computers & Operations Research, 38(1), 340-356, 2011.

[32] Günther ,O., Kulak, O., Kalayci ,C. B. ve Polat , O., A perturbation based variable neighborhood search heuristic for solving the vehicle routing problem with simultaneous pickup and delivery, European Journal of Operational Research, 242, 369-382, 2015.

[33] Junqueira, L., & Morabito, R. Heuristic algorithms for a three-dimensional loading capacitated vehicle routing problem in a carrier. Computers & Industrial Engineering, 88, 110–130, 2015.

[34] Pollaris, H., Braekers, K., Caris, A., Janssens, G. K., & Limbourg, S., Vehicle routing problems with loading constraints: state-of-the-art and future directions, OR Spectrum, 37(2), 297–330, 2015.