

A Heuristic Algorithm for Vehicle Routing Problems with Simultaneous Pick-Up and Delivery and Hard Time Windows

Suna Cetin^{1,2*}, Cevriye Gencer³

¹Department of Industrial Engineering, Faculty of Engineering, Kirikkale University, Kirikkale, Turkey

²Department of Industrial Engineering, Faculty of Engineering, International Turkmen-Turk University, Kosi-Ashgabat Turkmenistan

³Faculty of Engineering, Department of Industrial Engineering, Gazi University, Ankara, Turkey

Email: s_ozel71@yahoo.com

Received January 2015

Abstract

In this study, Vehicle Routing Problems with Simultaneous Pick-Up and Delivery and Hard Time Windows (VRPSPDHTW), the special case of vehicle routing problems is discussed. The goal of vehicle routing problems is generally, the minimization of travelled distance or travelling cost. In the literature, it is seen that same objective functions are defined in time windows vehicle routing problems. However, in time window vehicle routing problems, waits resulting from time window should be taken into account. In the study, objective function was specified as minimization of waits in VRPSPDHTW and the mathematical model has been defined as a set. In addition, heuristic algorithms are proposed for the solution of the problem. Solomon data set were modified to fit the structure of the problem and proposed algorithm was tested.

Keywords

Vehicle Routing Problems, Pick-Up and Delivery Service, Time Windows

1. Introduction

The problems concerning the distribution of goods between depots and final users are known as Vehicle Routing Problems (VRPs). Vehicle routing problems are widespread in distribution and logistics [1]. The general VRP comprised an assignment problem, travelling salesman problem and vehicle-path problem [2]. Dantzig and Ramser introduced VRP in 1959, and described a real world application, and proposed the first mathematical programming formulation and algorithmic approach for the solution of the problem [3].

The basic VRP involves a central depot, a set of customers requiring delivery of goods from the depot, and a fleet of identical vehicles located at the depot. The objective of VRP is to minimize the total traveled distance, traveling cost or the number of vehicles used to serve all customers. In the field of combinatorial optimization,

*Corresponding author.

the VRP is regarded as NP-hard. Several classes of VRPs, which are extensions of basic VRP, have been studied in the literature. Details and literature of VRP and its variants are given by [4] and [5].

The sub-class of VRP in which customers ask for delivery in a specified time window $[a, b]$ is referred to as vehicle routing problem with time windows constraints (VRPTW) [6]. Time windows characterize the earliest (a_i) and the latest (b_i) allowable times that the service should begin within. A vehicle is permitted to arrive earlier than a_i to a customer i , in which case it will wait until service begins at time a_i . Late arrivals, *i.e.* arrivals after b_i , are not permitted. The problems considering this type of time windows are called as vehicle routing problems with hard time windows (VRPHTW). On the other hand, both lower and upper bounds of time windows need not be met, but can be violated. These problems are known as vehicle routing problems with soft time windows (VRPSTW) [7]. Additionally, the total duration of any route (traveling time + service time) cannot exceed the pre-determined route duration. Salvensberg proved that the VRPTW is an NP problem [8]. Typical applications of the VRPTW include bank deliveries, school bus routing, postal deliveries [9]-[11]. The early work on VRPTW are generally case study oriented [12]-[14]. Later, researchers focused their attention on the development of effective optimal approaches and the design of heuristics for solving realistic-size problems. Detailed studies about VRP and VRPTW are [10] [11] [15] [16].

An extension of basic VRP is vehicle routing problems with pickup and delivery (VRPPD). In VRPPD, every customer has 2 two types of demand such as pick up demand (p_i) and delivery demand (d_i). Delivery demand refers to the amount of goods transported from depot to customer; pickup demand refers to the amount of goods transported from customer to depot. VRPPD is classified into 3 groups considering the dealing with delivery and pick up demand aspects as follows:

Delivery first- pickup second VRPPD: Customers are visited more than once. First, goods are transported from depot to the customers; then, from customers to depot. There is precedence between delivery and pick up customers.

Mixed pickups and deliveries VRPPD: There is not precedence between delivery and pick up. Delivery demand and pick up demand is serviced in a mixed sequence. Customers are visited more than once.

Simultaneous pickups and deliveries VRPPD: Delivery and pick up demand are serviced simultaneously. This type of problem is specifically denoted as VRPSPD. Customers are visited only once.

VRPSPD is encountered in real life. For example, in the soft drink industry full bottles are transported from depot to markets, empty bottles are returned back to the depots and in the grocery industry goods flow from depot to market while outdated products flow to depots. VRPSPD is an NP-hard combinatorial optimization problem because it is a version of VRP [17]. Min introduced VRPSPD and proposed an algorithm for transportation of books between libraries where there is one depot, and there are 2 vehicles and 22 customers [18]. That algorithm is based on cluster first-route second approach. Dethloff proposed a mathematical formulation for the problem and also developed an insertion-based heuristic to solve the problem [19]. In literature, mathematical model for VRPSPD is proposed 29 by [18]-[23]. Ropke and Pisinger developed a unified model capable of handling most variants of the problem from the literature [24]. Bianchessi and Righini presented and compared the performance of constructive algorithms, local search and tabu search algorithms [25]. Tang Montane proposed a mathematical formulation and a tabu search algorithm, which is intensified and diversified by the use of frequency penalization scheme for VRPSPD [26]. Zachariadis *et al.* proposed a hybrid meta-heuristic approach based on tabu search and guided local search [17]. Gajpal and Abad presented an ant colony system algorithm for VRPSPD [27]. Ai and Kachitvchyanukul proposed a mathematical formulation and a particle swarm optimization algorithm for VRPSPD [23]. The formulation is generalization of mathematical formulations proposed by Min [18], Dethloff [19], and Tang Montane *et al.* [26]. Chen *et al.* proposed a hybrid meta-heuristic method based on record to record travel tabu list and route improvement routines [28]. Zachariadis *et al.* presented an adaptive memory programming methodology algorithm for the VRPSPD [17]. Angellini and Mansini developed a branch and price algorithm for VRPSPD [29]. Mingyong and Erbao (2010) proposed an improved differential evolution algorithm (IDE) for solving VRPSPD [30]. Unlike the work of Angellini and Mingyong, in this study, the objective in determined problems is to minimize the simultaneous pick up and hard time windows and arp objective function waits. Mathematical model was specified and heuristic algorithm for the solution was developed.

2. Problem Description and Notation

We propose a mathematical formulation and develop a heuristic algorithm for the VRPSPDHTW. Proposed

problem is NP-hard as it is the extension of VRPSPD and VRPTW. The proposed mathematical model is based on Dethloff's mathematical model where time windows are inserted and objective function is considered as the minimization of total waiting time.

VRPSPDHTW can be formally defined as follows. Let $G = (V, A)$ is a graph where $V = \{v_0, v_1, \dots, v_n\}$ is a vertex set, and $A = \{(v_i, v_j) : v_i, v_j \in V, i \neq j\}$ is an arc set. Vertex v_0 represents the depot and vertex v_i represents the customers where $i = 1, \dots, n$. Every customer has a nonnegative pick up quantity p_i , delivery quantity d_i , and service time s_i . Each vehicle has a capacity of Q . The goal of VRPSPDHTW is to determine a set of routes that satisfy the following requirements [32]:

- 1) Each route starts and ends at the depot,
- 2) Each customer is serviced exactly once by exactly one vehicle,
- 3) All customer delivery and pick up demand are totally satisfied in the related time interval (a vehicle is permitted to arrive earlier than a_i to a customer but not permitted to arrive later than b_i)
- 4) Total vehicle load through any route does not exceed the vehicle capacity Q ,
- 5) Homogeneous fleet of vehicles exists in the depot.
- 6) Total waiting time is minimized.

Notations and parameters used in the model are given as:

J: Set of all customer locations

J_0 : Set of all nodes, *i.e.* customer locations and depot

V: Set of all vehicles

C: Vehicle capacity

t_{ij} : Travel time between nodes

s_i : Service time

Z_{ij} : Addition of service time and travel time between nodes ($Z_{ij} = s_i + t_{ij}$)

W_{iv} : Starting time of service at node i by vehicle v

V_{iv} : Arrival time at node i by vehicle v

D_j : Delivery amount of customer j

N: Number of nodes,

P_j : Pick up amount of customer,

M: A large number

a_i : The earliest starting time of service at node i

b_i : The latest starting time of service at node i

Decision Variables:

l'_v : Load of vehicle v when leaving the depot,

l_j : Load of vehicle v after having serviced customer j ,

π_j : Variable used to prohibit sub tours,

X_{ijv} = binary variable indicating where vehicle v travels directly from node i to node j ($x_{ijv} = 1$) or not ($x_{ijv} = 0$)

$$\text{Min}Z = \sum_{i \in J_0} \sum_{j \in J_0} \sum_{v \in V} T_i x_{ijv} \quad (1)$$

$$\sum_{i \in J_0} \sum_{v \in V} x_{ijv} = 1 \quad j = 1, 2, \dots, J \quad (2)$$

$$\sum_{i \in J_0} x_{isv} = \sum_{j \in J_0} x_{sjv} \quad v = 1, 2, \dots, V \quad (3)$$

$$l'_v = \sum_{i \in J_0} \sum_{j \in J} D_j x_{ijv} \quad v = 1, 2, \dots, V \quad (4)$$

$$l_j \geq l'_v - D_j + P_j - M(1 - x_{0jv}) \quad v = 1, 2, \dots, V; j = 1, 2, \dots, J \quad (5)$$

$$l_j \geq l_i - D_j + P_j - M(1 - \sum_{v \in V} x_{ijv}) \quad i, j = 0, 1, 2, \dots, J \quad (6)$$

$$l'_v \leq C_v \quad v = 1, 2, \dots, V \quad (7) \quad \text{and} \quad l_j \leq C_v \quad v = 1, 2, \dots, V; j = 1, 2, \dots, J \quad (8)$$

$$\pi_j \geq \pi_i + 1 - n \left(1 - \sum_{v \in V} x_{ijv} \right) \quad j = 1, 2, \dots, J; i = 1, 2, \dots, J \quad (9)$$

$$W_{iv} - W_{jv} + Mx_{ijv} \leq M - Z_{ij} \quad i, j = 0, 1, 2, \dots, J; v = 1, 2, \dots, V \quad (10)$$

$$a_j \sum_i x_{ijv} \leq W_{jv} \leq b_j \sum_i x_{ijv} \quad i, j = 0, 1, 2, \dots, J; v = 1, 2, \dots, V \quad (11)$$

$$V_{jv} \leq W_{iv} + Z_{ij}x_{ijv} + M(1 - x_{ijv}) \quad i, j = 0, 1, 2, \dots, J; v = 1, 2, \dots, V \quad (12)$$

$$V_{iv} \leq M - (M - Z_{oi})x_{0iv} \quad j = 0, 1, 2, \dots, J; v = 1, 2, \dots, V \quad (13)$$

$$W_{iv} \geq Z_{oi} - (1 - x_{0iv})M \quad j = 0, 1, 2, \dots, J; v = 1, 2, \dots, V \quad (14)$$

$$V_{iv} \leq W_{iv} \quad i = 0, 1, 2, \dots, J; v = 1, 2, \dots, V \quad (15)$$

$$T_j = \sum_v (W_{jv} - V_{jv}) \quad j = 0, 1, 2, \dots, J; v = 1, 2, \dots, V \quad (16)$$

$$\pi_j \geq 0 \quad j = 1, 2, \dots, J \quad (17)$$

$$\text{and } x_{ijv} \in \{0, 1\} \quad j = 1, 2, \dots, J; i = 1, 2, \dots, J; v = 1, 2, \dots, V \quad (18)$$

In the model; the objective function (1) minimizes the waiting times between the nodes, constraint (2) states that every customer node must be serviced exactly once, constraint (3) states that each customer must be serviced with the same vehicle. Constraint (4) gives the initial vehicle load, constraint (5) is the vehicle load after first customer, and constraint (6) is the vehicle load 'en route'. Constraint (7) and (8) compare vehicle load and vehicle capacity after first customer and 'en route'. Constraint (9) is a sub tour breaking constraint. Constraint (10) determines the starting time of service at all nodes through the route and constraint (11) assures that the starting time of service is between the time intervals. Constraint (12) and (13) determine the arrival time to any node through the route, and arrival time to the first node after the depot, respectively. Constraint (14) determines the starting time of service at the first node after the depot. Constraint (15) assures that the service will start after the vehicle arrives to the node, constraint (16) determines the waiting time. Constraint (17) and (18) describe the domain of decision variables: π_j are non-negative variables, and x_{ijv} are binary variables.

3. A Heuristic for VRSPDHTW

Due to the results of test problems solved using the mathematical model proposed above, it is clearly determined that this approach is not capable of yielding good results in a reasonable time for bigger sized test problems. In order to solve bigger sized problems, a heuristic approach is proposed in this paper. The step by step flow of the heuristic is given below:

1) Open a list of nodes L and sort them by increasing values of the earliest starting time of service a_i . If several nodes have the same a_i , sort them by increasing values of the latest starting time of service b_i .

2) Pick up the top node i on the list L and place it at the bottom of list K , then delete node i from the list L and calculate I_0 and I_q :

$$I_0 = \sum D_j \quad I_q = I_q - D_q + P_q \quad (19)$$

3) Calculate RD and RP values for nodes at list K where $RD(q)$ stands for *the* residual delivery capacity, and $RP(q)$ stands for the residual pick up capacity at the depot

$$RD(0) = C - \sum D_s \quad RD(q) = \min\{RD(PRI(q)), C - I_q\} \quad (20)$$

$$RP(0) = C - \sum P_s \quad RP(q) = \min\{RP(SUI(q)), C - I_q\}$$

4) Verify that the following formulation for nodes in list L is satisfied,

$$D(u) \leq RD(i) \quad \text{and} \quad P(u) \leq RP(i) \quad (21)$$

If it is satisfied, this means that node u can be placed after node i . If not, node u cannot be placed after node i , which means that it is infeasible.

5) Place the nodes that satisfy step 4 at list L' , if $L' = \emptyset$ close list K and return to step 2.

6) Calculate W_{ik} and V_{jk} values for nodes at List L' (V_{jk} , the arrival time at node j by vehicle k ; F_{ik} , the departure time from node i by vehicle k)

$$i = 0 \quad F_{0k} = 0 \quad i \neq 0 \quad \left\{ \begin{array}{ll} a_i \leq V_{ik} \leq b_i & F_{iv} = V_{ik} + s_i \\ V_{ik} < a_i & F_{iv} = a_i + s_i \\ V_{ik} > b_i & \text{inf } \textit{easible} \end{array} \right\} \text{ and } V_{jk} = F_{ik} + t_i \quad (22)$$

If $V_{ik} > b_i$ for all nodes at list L' then close list K , return to step 2.

7) Calculate the waiting times as

$$\begin{array}{ll} a_i \leq V_{ik} \leq b_i & W = 0 \\ V_{ik} < a_i & W = a_i - V_{ik} \end{array} \quad (23)$$

8) If $F_{nk} + t_{n0} > b_0$ (this means; the time to return to the depot is bigger than the latest starting time of service for depot), it is infeasible. If all nodes on list K' satisfy this condition, return to step 2

9) Choose the node that has smallest W_i and place on list K , delete the node from list L , return to step 2.

10) If List L is empty, stop the algorithm.

In order to investigate the performance of the mathematical model [31] and heuristic algorithm [32] proposed here, test problems with 5, 7, 10, 15 and 20 customers are solved. The main idea of choosing these problems is that for test problems with more customers the mathematical model cannot produce an optimal solution in a reasonable time period which is considered to be 7200 seconds in this study. The original test problems of Solomon includes 100 customers in each of the cases, satisfying the constraints in our case and the total capacity of the vehicles are determined to be half of its original value.

4. Numerical Examples

Since VRR_SPD_HTW has never been solved before, there is no data to test VRRSPDHTW in the literature. For this reason, Solomon's data is modified in order to be used in our algorithm. A full description of Solomon's problems can be found in Solomon [10]. Solomon's problems are grouped into 3 categories: C, R, and RC. Every category has 8 to 12 examples. Each example has a depot and 100 customers. Every category is classified into 2 groups as Type 1 and Type 2. The differences between Type1 problems and Type 2 problems are vehicle capacity and time windows. Type 1 problems have narrow time windows and small vehicle capacity while Type 2 problems have wide time windows and large vehicle capacity. The locations of customers are uniformly distributed in R1 and R2, and clustered in C1 and C2. For groups RC1 and RC2, the clustered and random distributions are mixed.

We use Salhi and Nagy's [20] formulations to modify Solomon's data. In order to investigate the effectiveness of both the mathematical model and the heuristic, test problems with 5, 7, 10, 15 and 20 customers are generated. The original test problems of Solomon includes 100 customers in each of the cases and in order to satisfy the constraints, the total capacity of the vehicles are determined to be half of its original value. The mathematical model is solved using a personal computer with a microprocessor with 2.6 Ghz and under 1 Gb Ram using GAMS package with a time limit of 7200 seconds. The solutions with the heuristic were calculated at the same computer using MATLAB coding language. The objective in determined problem is the minimization of the waiting times. The objective is, therefore, to expect the function take a value between 0 and + infinite. If in any problem the optimum solution value is $z = 0$, then it means that service is provided to the customers without making them wait. In comparing the results of mathematical model and heuristic algorithm, CPU average and Deviation average were used depending on the problem type. CPU average and Deviation average were calculated by dividing the sum of CPU and Deviation values into the number of the problems for every problem. Deviation values stand for the distance of z value of related problem in terms of percentage to the best z value. These values were calculated according to the given formula below:

$$\text{Deviation MM} = ((MMresult - \min(MMresult, SAresult)/MMresult)*100 \quad (24)$$

$$\text{Deviation SA} = ((SAresult - \min(MMresult, SAresult)/SAresult)*100$$

In this formula, MMresult represents MMathematical model results and SA result represents the results of heuristic algorithm. The obtained CPU and Deviation averages depending on the problem type can be seen in Cetin's desirtation [32] The Deviation Average = 0 in tables means that the related method gives the best result in all the problem types. Being different from zero gives how far it is from the best solution on average as percentage. In large-sized sample problem, heuristic algorithm gives better results when compared to mathematical model. This can be seen in Cetin's desirtation also [32].

5. Conclusion

In this study, VRSPDHTW is determined. For the determined problem, mathematical model and heuristic algorithm aiming to minimize the waiting times are developed. For sample groups with 5, 7, 10, 15 and 20 customers are examined. The demand of Solomon's Data are modified to get pickup and delivery demand. 112 problems were solved for each sample customer group. The results of mathematical model and heuristic algorithm were compared for sample groups by calculating deviation values depending on the problem type. When the results are analyzed, it was seen that heuristic algorithm obtains values very close to mathematical model in small-sized problems. In large-sized problems, however, it was observed that heuristic algorithm gives better results when compared to mathematical model. Moreover, it was observed that in problems where mathematical model cannot reach a solution, heuristic model reaches solution.

References

- [1] Prins, C., Labadi, N. and Reghioui, M. (2009) Tour Splitting Algorithms for Vehicle Routing Problems. *International Journal of Production Research*, **47**, 507-535. <http://dx.doi.org/10.1080/00207540802426599>
- [2] Wang, C.H. and Lu, J.Z. (2010) A Hybrid Genetic Algorithm That Optimizes Capacitated Vehicle Routing Problems. *Expert Systems with Applications*, **36**, 2921-2936. <http://dx.doi.org/10.1016/j.eswa.2008.01.072>
- [3] Dantzig, G.B. and Ramser, J.H. (1959) The Truck Dispatching Problem. *Management Science*, **6**, 80-91.
- [4] Toth, P. and Vigo, D. (2002) The Vehicle Routing Problem. Siam, Philadelphia. <http://dx.doi.org/10.1137/1.9780898718515>
- [5] Eksioglu, B., Vural A.V. and Reisman, A. (2009) The Vehicle Routing Problem: A Taxonomic Review. *Computers & Industrial Engineering*, **57**, 1472-1483. <http://dx.doi.org/10.1016/j.cie.2009.05.009>
- [6] Confeeore, G., Corini, D. and Stecca, G. (2008) A Computaional Method for Pricing of Delivery Service in a Logistics Network. *International Journal of Production Research*, **46**, 1231-1242. <http://dx.doi.org/10.1080/00207540701224285>
- [7] Calvete, H.I., Gale, C., Oliveros, M.J. and Sanchez-Valderde, B. (2007) A Goal Programming Approach to Vehicle Routing Problems with Soft Time Windows. *European Journal of Operational Research*, **177**, 1720-1733. <http://dx.doi.org/10.1016/j.ejor.2005.10.010>
- [8] Savelsbergh, M.W.P. (1985) Local Search in Routing Problems with Time Windows. *Annals of Operations Research*, **4**, 285-305. <http://dx.doi.org/10.1007/BF02022044>
- [9] Solomon, M.M. (1987) Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constrains. *Operations Research*, **35**, 254-265. <http://dx.doi.org/10.1287/opre.35.2.254>
- [10] Solomon, M.M. and Desrosier, J. (1988) Survey Paper—Time Constrained Routing and Scheduling Problems. *Transportation Science*, **22**, 1-13. <http://dx.doi.org/10.1287/trsc.22.1.1>
- [11] Pullen, H. and Webb, M. (1967) A Computer Application to a Transport Scheduling Problem. *Computer Journal*, **10**, 10-13. <http://dx.doi.org/10.1093/comjnl/10.1.10>
- [12] Knight, K. and Hofer, J. (1968) Vehicle Scheduling with Timed and Connected Calls: A Case Study. *Operational Research Quarterly*, **19**, 299-310.
- [13] Madsen, O.B.G. (1976) Optimal Scheduling of Trucks—A Routing Problem with Tight Due Times for Delivery. In: Strobel, H., Genser, R. and Etschmaier, M., Eds., *Optimization Applied to Transportation Systems*, IIASA, International Institute for Applied System Analysis, Luxemburg, 126-136.
- [14] Bouthillier, A. and Crainic, T.G. (2005) A Cooperative Parallel Meta Heuristic for the Vehicle Routing Problem with Time Windows. *Computers & Operations Research*, **32**, 1685-1708. <http://dx.doi.org/10.1016/j.cor.2003.11.023>
- [15] Dondo, R. and Cerda, J. (2007) A Cluster-Based Optimization Approach for the Multi-Depot Heterogeneous Fleet Vehicle Routing Problem with Time Windows. *European Journal of Operation Research*, **176**, 1478-1507. <http://dx.doi.org/10.1016/j.ejor.2004.07.077>

- [16] Zachariadis, E.E., Tarantilis, C.D. and Kiranoudis, C.T. (2009) Hybrid Metaheuristic Algorithm for the Vehicle Routing Problem with Simultaneous Delivery and Pick-Up Service. *Expert System with Applications*, **36**, 1070-1081. <http://dx.doi.org/10.1016/j.eswa.2007.11.005>
- [17] Min, H. (1989) The Multiple Vehicle Routing Problem with Simultaneous Delivery and Pick-Up Points. *Transportation Research Part A: General*, **23**, 377-386. [http://dx.doi.org/10.1016/0191-2607\(89\)90085-X](http://dx.doi.org/10.1016/0191-2607(89)90085-X)
- [18] Dethloff, J. (2001) Vehicle Routing and Reverse Logistics: The Vehicle Routing Problem with Simultaneous Delivery and Pick-Up. *OR-Spektrum*, **23**, 79-96. <http://dx.doi.org/10.1007/PL00013346>
- [19] Nagy, G. and Salhi, S. (2005) Heuristic Algorithms for Single and Multiple Depot Vehicle Routing Problems with Pickups and Deliveries. *European Journal of Operational Research*, **162**, 126-141. <http://dx.doi.org/10.1016/j.ejor.2002.11.003>
- [20] Montane, F.A.T. and Galvao, R.D. (2006) A Tabu Search Algorithm for the Vehicle Routing Problem with Simultaneous Pick-Up and Delivery Service. *Computers & Operations Research*, **33**, 595-619. <http://dx.doi.org/10.1016/j.cor.2004.07.009>
- [21] Dell'Amico, M., Righini, G. and Salani, M. (2006) A Branch-and-Price Approach to the Vehicle Routing Problem with Simultaneous Distribution and Collection. *Transportation Science*, **40**, 235-247. <http://dx.doi.org/10.1287/trsc.1050.0118>
- [22] Ai, T. and Kachitvichyanukul, V. (2009) A Particle Swarm Optimization for the Vehicle Routing Problem with Simultaneous Pickup and Delivery. *Computers & Operations Research*, **36**, 1693-1702. <http://dx.doi.org/10.1016/j.cor.2008.04.003>
- [23] Ropke, S. and Pisinger, D. (2006) An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery with Time Windows. *Transportation Science*, **40**, 455-472. <http://dx.doi.org/10.1287/trsc.1050.0135>
- [24] Bianchessi, N. and Righini, G. (2009) Heuristic Algorithms for the Vehicle Routing Problem with Simultaneous Pick-Up and Delivery. *Computers & Operations Research*, **34**, 578-594. <http://dx.doi.org/10.1016/j.cor.2005.03.014>
- [25] Tang, F.A. and Galvao, R.D. (2002) Vehicle Routing Problems with Simultaneous Pick-Up and Delivery Service. *Journal of the Operational Research Society of India*, **39**, 19-33.
- [26] Gajpal, Y. and Abad, P. (2009) An Ant Colony System (ACS) for Vehicle Routing Problem with Simultaneous Delivery and Pickup. *Computers & Operations Research*, **36**, 3215-3223. <http://dx.doi.org/10.1016/j.cor.2009.02.017>
- [27] Chen, A.-L., Yang, G.-K. and Wu, Z.-M. (2006) Hybrid Discrete Particle Swarm Optimization Algorithm for Capacitated Vehicle Routing Problem. *Journal of Zhejiang University Science A*, **7**, 607-614. <http://dx.doi.org/10.1631/jzus.2006.A0607>
- [28] Angellini, E. and Mansini, R. (2002) The Vehicle Routing Problems with Time Windows and Simultaneous Pick-Up and Delivery. In: Klose, A., Speranze, M.G. and Van Wassenhove, L.N., Eds., *Quantitative Approaches to Distribution Logistics and Supply Chain Management*, Springer, Berlin, Heidelberg, New York, 249-267. http://dx.doi.org/10.1007/978-3-642-56183-2_15
- [29] Minyong, L. and Erbao, C. (2010) An Improved Differential Evolution Algorithm for Vehicle Routing Problem with Simultaneous Pickups and Deliveries and Time Windows. *Engineering Application of Artificial Intelligence*, **23**, 188-195. <http://dx.doi.org/10.1016/j.engappai.2009.09.001>
- [30] Cetin, S. and Gencer, C. (2010) Vehicle Routing Problems with Simultaneous Pick-Up and Delivery and Hard Time Windows: Mathematical Model. *Journal of the Faculty of Engineering and Architecture of Gazi University*, **25**.
- [31] Cetin, S. (2011) Vehicle Routing Problems with Simultaneous Pick-Up and Delivery and Time Windows. PhD Thesis.