

PAPER • OPEN ACCESS

Discrete Optimization Model for Vehicle Routing Problem with Scheduling Side Cosntraints

To cite this article: Dedy Juliandri *et al* 2018 *IOP Conf. Ser.: Mater. Sci. Eng.* **300** 012024

View the [article online](#) for updates and enhancements.

You may also like

- [Multi-Compartment Vehicle Routing Problem to Find the Alternative Distribution Route of Petroleum Product Delivery](#)
I F Febriandini, Yuniaristanto and W Sutopo

- [Improvement Route for Distribution Solutions MDVRP \(Multi Depot Vehicle Routing Problem\) using Genetic Algorithm](#)
R Fitriana, P Moengin and U Kusumaningrum

- [A Survey for Vehicle Routing Problems and Its Derivatives](#)
Ming Han and Yabin Wang



The Electrochemical Society

Advancing solid state & electrochemical science & technology

DISCOVER
how sustainability
intersects with
electrochemistry & solid
state science research



Discrete Optimization Model for Vehicle Routing Problem with Scheduling Side Constraints

Dedy Juliandri¹, Herman Mawengkang², F. Bu'ulolo²

¹Graduate School of Mathematics, University of Sumatera Utara, Medan, Indonesia

²Department of Mathematics, University of Sumatera Utara, Medan, Indonesia

Email: hmawengkang@yahoo.com

Abstract. Vehicle Routing Problem (VRP) is an important element of many logistic systems which involve routing and scheduling of vehicles from a depot to a set of customers node. This is a hard combinatorial optimization problem with the objective to find an optimal set of routes used by a fleet of vehicles to serve the demands a set of customers. It is required that these vehicles return to the depot after serving customers' demand. The problem incorporates time windows, fleet and driver scheduling, pick-up and delivery in the planning horizon. The goal is to determine the scheduling of fleet and driver and routing policies of the vehicles. The objective is to minimize the overall costs of all routes over the planning horizon. We model the problem as a linear mixed integer program. We develop a combination of heuristics and exact method for solving the model.

Keywords: Integer programming, Transportation, Scheduling, Combined method

1. Introduction

Vehicle Routing Problem (VRP) is one of the important issues that exist in transportation system. This is a well known combinatorial optimization problem which consists of a customer population with deterministic demands, and a central depot which acts as the base of a homogeneous fleet of vehicles. The objective is to design a set of vehicle routes starting and terminating at the central depot, such that the demand of customers is totally satisfied, each customer is visited once by a single vehicle, the total demand of the customers assigned to a route does not exceed vehicle capacity, and to minimize the overall travel cost, taking into account various operational constraints.

VRP was first introduced by Dantzig and Ramser (1959). Due to the model has a lot applications, particularly in logistic system, and the combinatorial nature contained in the structure of the problem, it is not surprising that many researchers have been working in this area to discover new methodologies in order to solve the problems efficiently. There are a number of survey can be found in literature for VRP, such as, Caceres-Cruz et al (2014), Braysy and Gendreau (2005), Cordeau et al. (2007), Golden et al. (2002), Laporte and Sement (2002), and books (Golden et al. (2008), Toth and Vigo (2002)).

Mathematically, VRP can be defined as follows: vehicles with a fixed capacity Q must deliver order quantities nonnegative q_i ($i = 1, \dots, n$) of goods to n customers from a single depot ($i = 0$). These vehicles must come back to the depot after serving customers. Knowing the distance d_{ij} between customers i and j ($i, j = 1, \dots, n$), the objective of the problem is to minimize the total distance traveled



by the vehicles in a way that only one vehicle handles the deliveries for a given customer and the total quantity of goods that a single vehicle delivers is not larger than Q .

There are three types of approach have been proposed for solving VRP, viz., exact, heuristics, and hybrid (a combined between heuristics and exact method).

Bettinelli et al. (2011) propose an exact method called branch-and-cut-and-price for solving VRP. Another exact method called column generation was used by Ceselli et al. (2009) and Goel (2010). Kok et al. (2010) propose dynamic programming as an exact method for solving VRP. More exact solution for VRP can be found in Baldacci et al (2010), and Baldacci et al. (2012).

Due to NP-hardness of VRP, some meta-heuristics are developed to solve it such as genetic algorithms (Cheng and Wang, 2009; Ursani et al., 2011; Vidal et al., 2013), ant colony (Ding et al., 2012; Yu and Yang, 2011), Tabu search (Belhaiza et al., 2013; Ho and Haugland, 2004), simulated annealing (Ban˜ os et al., 2013; Deng et al., 2009; Kuo, 2010; Tavakkoli-Moghaddam et al., 2007, 2011). Hybrid approach for solving VRP can be found in Guimarans (2012).

Pickup and delivery problems (PDPs) are a class of vehicle routing problems in which objects or people have to be transported between an origin and a destination. They can be classified into three different groups. The first group consists of many-to-many problems, in which any vertex can serve as a source or as a destination for any commodity. An example of a many-to-many problem is the Swapping Problem (Anily and Hassin, 1992). In this problem, every vertex may initially contain an object of a known type of commodity as well as a desired type of commodity. The problem consists of constructing a route performing the pickups and deliveries of the objects in such a way that at the end of the route, every vertex possesses an object of the desired type of commodity. Problems in the second group are called one-to-many-to-one problems. In these problems commodities are initially available at the depot and are destined to the customer vertices; in addition, commodities available at the customers are destined to the depot. Finally, in one-to-one problems, each commodity (which can be seen as a request) has a given origin and a given destination. Problems of this type arise, for example, in courier operations and door-to-door transportation services.

The existing literature on the VRP time windows with pick-up and delivery mainly (VRPTWPD) focuses on heuristic algorithms (see e.g., Nanry and Barnes (2000), Rˆopke and Pisinger (2006)). However, there are also a few studies that have introduced exact solution algorithms (see, e.g., Dumas et al. (1991), Savelsbergh and Sol (1995), Rˆopke and Cordeau (2009)).

In this paper we add the scheduling of fleet and driver in the VRPTWPD. The problem can be modeled as a mixed integer linear program. We use a direct search method for solving the problem.

2. VRP Problem formulation

Before we model the VRPTWPD, it is necessary to model the VRP as a framework to model the advanced problem. Let a graph $G(V, A)$ is given with nodes $V = C \cup \{0\}$ and arcs A , in VRP C is a representation of the set of customers, which 0 is the depot. Moreover, we have a set R of resources which e.g. can be load and/or time. Each resource $r \in R$ has a resource window $[a_i^r, b_i^r]$ that must be met upon arrival to node $i \in V$, and a consumption $t_{ij}^r \geq 0$ for using arc $(i, j) \in A$. A resource consumption at a node $i \in C$ is modeled by a resource consumption at edge (i, j) , and hence usually $t_{0j}^r \geq 0$ for all $j \in C$. A global capacity limit Q can be modeled by imposing a resource window $[0, Q]$ for the depot node 0.

The VRP can now be stated as: Find a set of routes starting and ending at the depot node 0 satisfying all resource windows, such that the cost is minimized and all customers C are visited. A solution to the VRP will consist of a number of routes which starts from depot and at the end will be back to the depot.

In the following let c_{ij} be the cost of arc $(i, j) \in A$, x_{ij} be the binary variable indicating the use of arc $(i, j) \in A$, and t_{ij}^r (the resource stamp) be the consumption of resource $r \in R$ at the beginning of arc $(i, j) \in A$. Let $\delta^+(i)$ and $\delta^-(i)$ be the set of outgoing respectively ingoing arcs of node $i \in V$. Combining the two index model from Bard et al. [3] with the constraints ensuring the time windows for the

asymmetric travelling salesman by Ascheuer et al. [1]. A mathematical model can be formulated as follows:

$$\min \sum_{(i,j) \in A} c_{ij} x_{ij} \quad (1)$$

$$\text{s.t.} \quad \sum_{(s,j) \in \delta^+(i)} x_{ij} = 1 \quad \forall i \in C \quad (2)$$

$$\sum_{(s,j) \in \delta^-(i)} x_{ij} = \sum_{(s,j) \in \delta^+(i)} x_{ij} \quad \forall i \in V \quad (3)$$

$$\sum_{(s,j) \in \delta^-(i)} (T_{ji}^r + t_{ji}^r x_{ij}) \leq \sum_{(s,j) \in \delta^+(i)} T_{ji}^r \quad \forall r \in R, \forall i \in C \quad (4)$$

$$a_i^r x_{ij} \leq T_{ji}^r \leq b_i^r x_{ij} \quad \forall r \in R, \forall (i,j) \in A \quad (5)$$

$$T_{ij}^r \geq 0 \quad \forall r \in R, \forall (i,j) \in A \quad (6)$$

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

The objective (1) sums up the cost of the used arcs. Constraints (2) ensure that each customer is visited exactly once, and (3) are the flow conservation constraints. Constraints (4) and (5) ensure the resource windows are satisfied. It is assumed that the bounds on the depot are always satisfied. Note, that no sub-tours can be present since only one resource stamp per arc exists and the arc weights are positive for all $(i,j) \in A : i \in C$.

For a one dimensional resource such as *load* a stronger lower bound of the LP relaxation can be obtained by replacing (4) to (6) with $\sum_{(i,j) \in \delta^+(S)} x_{ij} \geq r(S)$, where $r(S)$ is a minimum number of vehicles needed to service the set S . All though this model can not be directly solved it is possible to overcome this problem by only including the constraints that are violated.

3. Mathematical Formulation of VRPFDPTW

To formulate the model, firstly we should refer to the formulation of VRP given in Section 2. Now we add another formulation regarding to the fleet and driver scheduling and also pick-up and delivery. Let T is the planning horizon and D is the set of drivers. The set of workdays for driver $l \in D$ is denoted by $T_l \subseteq T$. The start working time and latest ending time for driver $l \in D$ on day $t \in T$ are given by g_l^t and h_l^t , respectively. For each driver $l \in D$, let H denote the maximum weekly working duration. We denote the maximum elapsed driving time without break by F and the duration of a break by G .

Let K denote the set of vehicles. For each vehicle $k \in K$, let Q_k and P_k denote the capacity in weight and in volume, respectively. We assume the number of vehicles equals to the number of drivers. Denote the set of n customers (nodes) by $N = \{1, 2, \dots, n\}$. Denote the depot by $\{0, n+1\}$. Each vehicle starts from depot, $\{0\}$, and terminates at depot, $\{0\}$. Each customer $i \in N$ specifies a set of days to be visited, denoted by $T_i \subseteq T$. On each day $t \in T_i$, customer $i \in N$ requests service with demand of q_i^t in weight and p_i^t in volume, service duration d_i^t and time window $[a_i, b_i]$. Note that, for the depot on day t , we set $q_i^t = p_i^t = d_i^t = 0$.

We define binary variable x_{ijk}^t to be 1 if vehicle k travels from node i to j on day t , binary variable w_i^t to be 1 if customer i is not visited by a preferred vehicle on day t . Variable v_{ik}^t is the time that vehicle

k visits node i on day t . Binary variable z_{ik}^t indicates whether vehicle k takes a break after serving customer i on day t . Variable u_{ik}^t is the elapsed driving time for vehicle k at customer i after the previous break on day t . Binary variable y_{lk}^t is set to 1 if vehicle k is assigned to driver l on day t . Variables r_l^t and s_l^t are the total working duration and the total travel time for driver l on day t , respectively.

We define the notations to be used as follows.

Set:

T	The set of workdays in the planning horizon,
D	The set of drivers $D = D_I \cup D_E$,
T_l	The set of workdays for driver $l \in D$,
K	The set of vehicles,
N	The set of customers,
N_0	The set of customers and depot $N_0 = \{0\} \cup N$,
K_i	The set of preferable vehicles for customer $i \in N$,
T_i	The set of days on which customer $i \in N$ orders,

Parameter:

Q_k	The weight capacity of vehicle $k \in K$,
P_k	The volume capacity of vehicle $k \in K$,
c_{ij}	The travel cost (time) from node $i \in N_0$ to node $j \in N_0$,
cd_{lk}	The cost for driver $l \in D$ for vehicle $k \in K$,
$[a_i, b_i]$	The earliest and the latest visit time at node $i \in N_0$,
d_i^t	The service time of node $i \in N_0$ on day $t \in T_i$,
q_i^t	The weight demand of node $i \in N_0$ on day $t \in T_i$,
p_i^t	The volume demand of node $i \in N_0$ on day $t \in T_i$,
$[g_l^t, h_l^t]$	The start time and the latest ending time of driver $l \in D$ on day $t \in T$,
α_i^t	Pick up quantity for customer i on day $t \in T_i$,
β_i^t	Delivery quantity for customer i on day $t \in T_i$,
F	The maximum elapsed driving time without break,
G	The duration of break for drivers,

Variables:

x_{ilk}^t	Binary variable indicating whether vehicle $k \in K$ travels from node $i \in N_0$ to $j \in N_0$ on day $t \in T$,
v_{ik}^t	The time at which vehicle $k \in K$ starts service at node $i \in N_0$ on day $t \in T$,
z_{ik}^t	Binary variable indicating whether vehicle $k \in K$ takes break after serving node $i \in N_0$ on day $t \in T$,
u_{ik}^t	The elapsed driving time of vehicle $k \in K$ at node $i \in N_0$ after the previous break on day $t \in T$,
y_{lk}^t	Binary variable indicating whether vehicle $k \in K$ is assigned to driver $l \in D$ on day $t \in T$,

- r_l^t The total working duration of driver $l \in D$ on day $t \in T$,
- s_l^t The total travel distance of driver $l \in D$ on day $t \in T$,
- θ_{jk}^t Number of pick up demand of customer j served by vehicle $k \in K$ on day $t \in T$
- σ_{jk}^t Number of delivery demands of customer j served by vehicle $k \in K$ on day $t \in T$

The problem can be presented as a mixed integer linear programming model.

The objective of the problem is to minimize costs, which consists of the travel cost from depot to customers, travel cost from customer to customer, and the cost for drivers. Mathematically the objective can be written as

$$\text{Minimize } \sum_{j \in N} \sum_{k \in K} \sum_{t \in T} c_{0jk}^t x_{ojk}^t + \sum_{i \in N} \sum_{j \in N, j > i} \sum_{k \in K} \sum_{t \in T} c_{ijk}^t + \sum_{l \in D} \sum_{k \in K} \sum_{t \in T} cd_{lk}^t y_{lk}^t \quad (8)$$

Subject to:

$$\sum_{j \in N} x_{ji}^t = 1 \quad \forall i \in N_0, t \in T \quad (9)$$

$$\sum_{i \in N_0} x_{ij}^t = 1 \quad \forall j \in N, t \in T \quad (10)$$

$$\sum_{i \in N} \sum_{j \in N_0} q_i^t x_{ijk}^t \leq Q_k \quad \forall k \in K, t \in T \quad (11)$$

$$\sum_{i \in N} \sum_{j \in N_0} p_i^t x_{ijk}^t \leq P_k \quad \forall k \in K, t \in T \quad (12)$$

$$u_{jk}^t \geq u_{ik}^t + c_{ij} - M(1 - x_{ijk}^t) - Mz_{ik}^t \quad \forall i, j \in N_0, k \in K, t \in T \quad (13)$$

$$u_{jk}^t \geq c_{ij} - M(1 - x_{ijk}^t) \quad \forall i, j \in N, k \in K, t \in T \quad (14)$$

$$u_{ik}^t + \sum_{j \in N_0} c_{ij} x_{ijk}^t - F \leq Mz_{ik}^t \quad \forall i \in N_0, k \in K, t \in T \quad (15)$$

$$b_i \geq v_{ik}^t \geq a_i \quad \forall i \in N, k \in K, t \in T_i \quad (16)$$

$$v_{0k}^t \geq \sum_{l \in D} (g_l^t \cdot y_{lk}^t) \quad \forall k \in K, t \in T \quad (17)$$

$$v_{n+1,k}^t \leq \sum_{l \in D} (h_l^t \cdot y_{lk}^t) \quad \forall k \in K, t \in T \quad (18)$$

$$s_l^t \geq \sum_{i \in N_0} \sum_{j \in N_0} c_{ij} x_{ijk}^t - M(1 - y_{lk}^t) \quad \forall l \in D, k \in K, t \in T_i \quad (19)$$

$$r_l^t \geq v_{n+1,k}^t - g_l^t - M(1 - y_{lk}^t) \quad \forall l \in D, k \in K, t \in T_i \quad (20)$$

$$\sum_{k \in K} \theta_{jk}^t = \alpha_j^t \quad \forall j \in N, t \in T \quad (21)$$

$$\sum_{k \in K} \sigma_{jk}^t = \beta_j^t \quad \forall j \in N, t \in T \quad (22)$$

$$x_{ijk}^t, w_i^t, z_{ik}^t, y_{lk}^t \in \{0, 1\} \quad \forall i, j \in N_0, l \in D, k \in K, t \in T \quad (23)$$

$$v_{ik}^t, u_{ik}^t, r_l^t, s_l^t \geq 0 \quad \forall i, j \in N_0, l \in D, k \in K, t \in T \quad (24)$$

$$\theta_{jk}^t, \sigma_{jk}^t \in \{0, 1, 2, \dots\} \quad \forall j \in N, k \in K, t \in T \quad (25)$$

Constraints (9-10) are to ensure that only one customer is to be visited. Constraints (11-12) guarantee that the vehicle capacities are respected in both weight and volume. Constraints (13-14) define the elapsed driving time. More specifically, for the vehicle (k) travelling from customer i to j on day t , the elapsed driving time at j equals the elapsed driving time at i plus the driving time from i to j (i.e., $u_{jk}^t \geq u_{ik}^t + c_{ij}$) if the vehicle does not take a break at customer i (i.e., $z_{ik}^t = 0$); Otherwise, if the vehicle takes a break at customer i (i.e., $z_{ik}^t = 1$), the elapsed driving time at j will be constrained by (1) which make sure it is greater than or equal to the travel time between i and j (i.e., $u_{jk}^t \geq c_{ij}$). Constraints (15) guarantee that the elapsed driving time never exceeds an upper limit F by imposing a break at customer i (i.e., $z_{ik}^t = 1$) if driving from customer i to its successor results in a elapsed driving time greater than F . Constraints (16) make sure the services start within the customers' time window.

Constraints (17-18) describe that the starting time and ending time of each route must be between the start working time and latest ending time of the assigned driver. Constraint (19) is to calculate the total travel time for each assigned driver. Constraints (20) are necessary to define the working duration for each assigned driver on every workday. Constraints (21 – 22) define the pick up and delivery for each customer. Constraints (23-25) define the binary and other variables used in this formulation.

4. The Algorithm

Let

$$x = [x] + f, \quad 0 \leq f \leq 1$$

be the (continuous) solution of the relaxed linear programming problem, $[x]$ is the integer component of non-integer basic variable x and f is the fractional component.

There are three stages.

The first stage.

Step 1. Get row i^* such that $\delta_{i^*} = \min\{f_i, 1 - f_i\}$

(This is to minimize the deterioration of the objective function, and clearly corresponds to the integer basic infeasibility).

Step 2. Obtain

$$v_{i^*}^T = e_{i^*}^T B^{-1}$$

Step 3. Calculate the maximum movement of nonbasic j , using $\sigma_{ij} = v_{i^*}^T \alpha_j$

With j corresponds to

$$\min_j \left\{ \left| \frac{d_j}{\alpha_{ij}} \right| \right\}$$

Otherwise go to next non-integer nonbasic or superbasic j (if available). Eventually the column j^* is to be increased from LB or decreased from UB. If none go to next i^* .

Step 4.

Solve $B\alpha_{j^*} = \alpha_{j^*}$ for α_{j^*}

Step 5. Do ratio test for the basic variables due to the releasing of nonbasic j^* from its bounds.

Step 6. Perform the exchange basis

Step 7. If row $i^* = \{\emptyset\}$ go to Stage 2, otherwise

Repeat from step 1.

Stage 2. Pass1 : perform a movement of integer infeasible superbasics by fractional steps to obtain the complete integer feasibility.

Pass2 : do a heuristic for the integer feasible superbasics. The objective of this phase is to conduct a highly localized neighbourhood search to verify local optimality.

5. Conclusions

This paper is to develop efficient technique for solving one of the most economic important problems in optimizing transportation and distribution systems. The aim of this paper is to develop a model of vehicle routing with Time Windows, Fleet and Driver Scheduling, Pick-up and Delivery Problem This problem has additional constraints which are the limitation in the number of vehicles. The proposed algorithm employs nearest neighbor heuristic algorithm for solving the model.

References

- [1] B. L. Golden, A. A. Assad, and E. A. Wasil, "Routing Vehicles in the real world: Applications in the solid waste, beverage, food, dairy, and newspaper industries". In P. Toth and D. Vigo, editors, *The Vehicle Routing Problem*, pages 245-286. SIAM, Philadelphia, PA, 2002.
- [2] B. L. Golden, S. Raghavan, and E. A. Wasil, "The vehicle routing problem: latest advances and new challenges". Springer, 2008.
- [3] G. Laporte and F. Sement. Classical Heuristics for the Vehicle Routing Problem. In Toth, P. and Vigo, D., editors, *The vehicle Routing Problem*, SIAM Monographs on Discrete Matheamtics and Applications, pages 109-128. SIAM, Philadelphia, PA, 2002.
- [4] J.-F. Cordeau, G. Laporte, M.W.F. Savelsbergh, and D. Vigo. Vehivle Routing. In Barnhart, C. and Laporte, G., edotirs, *Transportation*, handbooks in Operation Research and Management Science, Pages 367-428. North-Holland, Amsterdam, 2007.
- [5] O. Bräysy and M. Gendreau. "Vehicle Routing Problem with Times Windows, Part II: Metaheuristics". *Transportation Science*, 39(1): 104-118, 2005.
- [6] P. Toth and D. Vigo. "The Vehicle Routing Problem Society for Industrial and Applied Matheamtics", Philadelphia, PA, 2002.
- [7] G.B. Dantzig and J.H. Ramser, "The truck dispatching problem", *Management Science*, Vol. 6, 1959, 80.
- [8] A. Bettinelli, A. Ceselli, and G. Righini. 2011. A branch-and-cut-and-price algorithm for the multi-depot heterogeneous vehicle routing problem with time windows. *Transportation Research Part C: Emerging Technologies* 19, 5 (2011), 723–740.
- [9] Ceselli, G. Righini, and M. Salani. 2009. A column generation algorithm for a rich vehicle-routing problem. *Transportation Science* 43, 1 (2009), 56–69.
- [10] A. Goel. 2010. A column generation heuristic for the general vehicle routing problem. In *Learning and Intelligent Optimization*, C. Blum and R. Battiti (Eds.). Lecture Notes in Computer Science, Vol. 6073. Springer, 1–9.
- [11] A.L. Kok, C.M. Meyer, H. Kopfer, and J.M.J. Schutten. 2010. A dynamic programming heuristic for the vehicle routing problem with time windows and European community social legislation. *Transportation Science* 44, 4 (2010), 442–454.
- [12] R. Baldacci, E. Bartolini, A. Mingozzi, and R. Roberti. 2010. An exact solution framework for a broad class of vehicle routing problems. *Computational Management Science* 7, 3 (2010), 229–268.
- [13] R. Baldacci, A. Mingozzi, and R. Roberti. 2012. Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. *European Journal of Operational Research* 218, 1 (2012), 1–6.
- [14] Cheng, C.B., Wang, K.P., 2009. Solving a vehicle routing problem with time windows by a decomposition technique and a genetic algorithm. *Expert Syst. Appl.* 36 (4), 7758–7763.
- [15] Ursani, Z., Essam, D., Cornforth, D., Stocker, R., 2011. Localized genetic algorithm for vehicle routing problem with time windows. *Appl. Soft Comput.* 11 (8), 5375–5390.
- [16] Vidal, T., Crainic, T.G., Gendreau, M., Prins, C., 2013. A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time windows. *Comput. Oper. Res.* 40 (1), 475–489.
- [17] Ding, Q., Hu, X., Sun, L., Wang, Y., 2012. An improved ant colony optimization and its application to vehicle routing problem with time windows. *Neurocomputing* 98 (3), 101–107.

- [18] Yu, B., Yang, Z.Z., 2011. An ant colony optimization model: the period vehicle routing problem with time windows. *Transp. Res. Part E: Logist. Transp. Rev.* 47 (2), 166–181.
- [19] Belhaiza, S., Hansen, P., Laporte, G., 2013. A hybrid variable neighborhood Tabu search heuristic for the vehicle routing problem with multiple time windows. *Comput. Oper. Res.* (in press), <http://dx.doi.org/10.1016/j.cor.2013.08.010>.
- [20] Ho, S.C., Haugland, D., 2004. A Tabu search heuristic for the vehicle routing problems with time windows and split deliveries. *Comput. Oper. Res.* 31 (12), 1947–1964.
- [21] Ban˜ os, R., Ortega, J., Gil, C., Fern´andez, A., Toro, F., 2013. A simulated annealing based parallel multi-objective approach to vehicle routing problems with time windows. *Expert Syst. Appl.* 40 (5), 1696–1707.
- [22] Deng, A.M., Mau, C., Zhou, Y.T., 2009. Optimizing research of an improved simulated annealing algorithm to soft time windows vehicle routing problem with pick-up and delivery. *Syst. Eng. Theory Pract.* 29 (5), 186–192.
- [23] Kuo, Y., 2010. Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem. *Comput. Ind. Eng.* 59 (1), 157–165.
- [24] Tavakkoli-Moghaddam, R., Gazanfari, M., Alinaghian, M., Salamatbakhsh, A., Norouzi, N., 2011. A new mathematical model for a competitive vehicle routing problems with time windows solved by simulated annealing. *J. Manuf. Syst.* 30 (2), 83–92.
- [25] D. Guimarans. 2012. Hybrid algorithms for solving routing problems. Ph.D. Dissertation. Universitat Aut`onoma Barcelona.
- [26] Nanry, W, W Barnes. 2000. Solving the pickup and delivery problem with time windows using reactive tabu search. *Transportation Research Part B: Methodological* 34(2) 107–121.
- [27] Røpke, S, D Pisinger. 2006. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science* 40(4) 455–472.
- [28] Røpke, S, J-F Cordeau. 2009. Branch and cut and price for the pickup and delivery problem with time windows. *Transportation Science* 43(3) 267–286.
- [29] Dumas, Y, J Desrosiers, F Soumis. 1991. The pickup and delivery problem with time windows. *European Journal of Operational Research* 54(1) 7–22
- [30] Savelsbergh, M, M Sol. 1995. The general pickup and delivery problem. *Transportation Science* 29(1) 17–29.