Construction of an optimal solution for a Real-World Routing-Scheduling-Loading Problem

José F. Delgado O.[1], Laura Cruz-Reyes[1], Juan J. González B.[1], Héctor Fraire H.[1], Rodolfo A. Pazos R.[2]

[1] Instituto Tecnológico de Ciudad Madero, México (ITCM)
[2] Centro de Nacional de Investigación y Desarrollo Tecnológico, México (CENIDET)
Email: francisco.delgado.orta@gmail.com, lcruzreyes@prodigy.net.mx, jjgonzalezbarbosa@hotmail.com, hfraire@prodigy.net.mx, pazos@sd-cenidet.com.mx

(Paper received on February 29, 2008, accepted on April 15, 2008)

Abstract. This work presents an exact method for the Routing-Loading-Scheduling Problem (RoSLoP). The objective of RoSLoP consists of optimizing the delivery process of bottled products in a company study case. RoSLoP, formulated through the well-known Vehicle Routing Problem (VRP), has been solved as a rich VRP variant through approximate methods. The exact method uses a linear transformation function, which allows the reduction of the complexity of the problem to an integer programming problem. The optimal solution to this method establishes metrics of performance for approximate methods, which reach an efficiency of 100% in distance traveled and 75% in vehicles used, objectives of VRP. The transformation function reduces the computation time from 55 to four seconds. These results demonstrate the advantages of the modeling mathematical to reduce the dimensionality of problems NP-hard, which permits to obtain an optimal solution of RoSLoP. This modeling can be applied to get optimal solutions for real-world problems.

Keywords: Optimization, Routing–Scheduling–Loading Problem (RoSLoP), Vehicle Routing Problem (VRP), rich VRP.

1 Introduction

The distribution and delivery processes are inherent to many manufacturing companies; in other cases, it is the main function of several service businesses. Though this could be considered in consequential, however, merchandise delivery in due time with the minimum quantity of resources, reduces operation costs, yielding savings between 5 to 20 % in total costs of products [1].

In recent years, many researchers have approached transportation problems based on real situations in two ways: formulating rich models of solution and developing efficient algorithms to solve them. RoSLoP, defined in [2] and extended in [3], is a high-complexity problem due its dimensionality.

RoSLoP formulation, associated with the transportation of bottled products in a company located in north eastern Mexico, satisfies the needs of the logistics group of the company. The application of a meta-heuristic algorithm based on an ant colony system (presented in [3]) to the RoSLoP problem permits to generate feasible solutions. However, performance metrics that measure the quality of the obtained solutions have not been created for this method. This work presents a new formulation for RoSLoP, based on reported methods in the literature for VRP variants, which uses a mathematical artifice that permits reducing the dimensionality of the problem and its solution as an integer programming problem. Therefore, the solution obtained is used as a measure of performance for heuristic algorithms.

This paper shows the exact method based on the solution of 12 VRP variants: CVRP, VRPM, HVRP, VRPTW, SDVRP, sdVRP, VRPMTW, OVRP, sdVRP, CCVRP, DDVRP and MDVRP. These variants and the sate of art of with rich VRP variants are described in section 2 and 3. Sections 4 and 5 are devoted to describe RoSLoP and the exact method. Section 6 show the experimentation with real-world instances; and section 7 presents the conclusions for future applications of this work.

2 The Vehicle Routing Problem (VRP)

VRP, defined by Dantzig in [4], is a classic problem of combinatorial optimization. It consists in one or various depots, a fleet of m available vehicles and a set of n customers to be visited, joined through a graph $G(V,E)$, where:

$V=\{v_0, v_1, v_2, \ldots, v_n\}$ is the set of vertex v_i such that, v_0 is the depot and the rest of the vertex represent the customers; each customer has a demand q_i of goods to be satisfied by the depot.

$E=\{(v_i, v_j) \mid v_i,v_j \varepsilon V, i \neq j\}$ is the set of edges where each edge has an associated value c_{ij} that represents the transportation cost from v_i to v_j.

The VRP consists of obtaining a set R of routes with a total minimum cost such that: each route starts and ends at the depot, each vertex $v_i \in V-\{v_0\}$ is visited only once by a route and the length of each route must be less than or equal to L. So, the main objective is to obtain a configuration with the minimum quantity of vehicles and traveled distance for satisfying all the customer demands.

2.1 Variants of VRP

The most known variants of VRP add several constraints to the basic VRP such as capacity of the vehicles (CVRP) [4], independent service schedules at the customers facilities (VRPTW-VRPMTW) [5], multiple depots to satisfy the demands (MDVRP) [6], customers to be satisfied by different vehicles (SDVRP) [7], a set of available vehicles to satisfy the orders (sdVRP) [8], customers that can ask and return goods to the depot (VRPPD) [9], dynamic facilities (DVRP) [10], line-haul and back-haul orders (VRPB) [11], stochastic demands and schedules (SVRP) [8], multiple use of the vehicles (VRPM) [12], a heterogeneous fleet to delivery the orders (HVRP) [13], orders to be satisfied in several days (PVRP) [6], constrained capacities of the customers for docking and loading the vehicles (CCVRP) [3], transit restrictions on the roads (rdVRP) [3], depots that can ask for goods to another depots (DDVRP) [3] and vehicles that can end its travel in several facilities (OVRP). A rich VRP variant, defined in [1] as an *Extended Vehicle Routing Problem*, is an application of VRP for real transportation problems. It is based on the Dantzig's formulation; however, it requires the addition of restrictions that represent the combination of many variants in a problem; which increases its complexity, making more difficult the computation of an optimal solution through exact algorithms.

3 Related works of rich VRP variants

Recent works have approached the solution of rich VRP problems like the DOMinant Project [14], which solves five variants of VRP in a transportation problem of goods among industrial facilities located in Norway. Goel [15] solves four VRP variants in a problem of sending packages for several companies. Pisinger [16] and Cano [17] solve transportation problems with five VRP variants.

RoSLoP was formulated initially in [18] with six VRP variants. Due a requirement of the company it was necessary to formulate a VRP with 11 variants in [19]. This new formulation allows the solution of instances of 12 VRP variants. Table 1 details the variants solved by various authors.

Table 1. Related works about known rich VRP variants

Solved variants / Autor	CVRP	VRPTW	VRPMTW	OVRP	VRPPD	MDVRP	SDVRP	sdVRP	VRPM	HVRP	CCVRP	DDVRP	rdVRP
Hasle [14]	✓	✓	✓						✓	✓			
Goel [15]	✓	✓							✓	✓			
Pisinger [16]	✓	✓		✓		✓		✓					
Cano [17]	✓	✓					✓		✓	✓			
Cruz et al. [2]	✓	✓				✓	✓		✓	✓			
Cruz et al. [3]	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓
This work	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓

A study of complexity factors of the base case of the problem is presented in Table 2. Rangel's mathematical formulation [18] requires 2^6 integer variables to solve 30 restrictions. Herrera's approach [19] needs 2^9 integer variables to solve 30 restrictions of the formulation. The proposed formulation contains 2^2 integer variables and 15 restrictions to solve 12 VRP variants.

Table 2. Complexity of the mathematical models created to RoSLoP

Complexity Elements / Method	Number of integer variables	Number of Restrictions	Solved VRP variants	Solver
Rangel [18]	2^6	30	6	Heuristic
Herrera [19]	2^9	30	11	Heuristic
This work	2^2	15	12	Exact

4 Definition of RoSLoP

RoSLoP, immersed in the logistics activity of the company study case, involves a subset of three tasks: routing, scheduling and loading. The mathematical model of RoSLoP was formulated with two classical problems: routing and scheduling through VRP and the loading through the Bin Packing Problem (BPP). Fig. 1 shows RoSLoP and its relation with VRP-BPP.

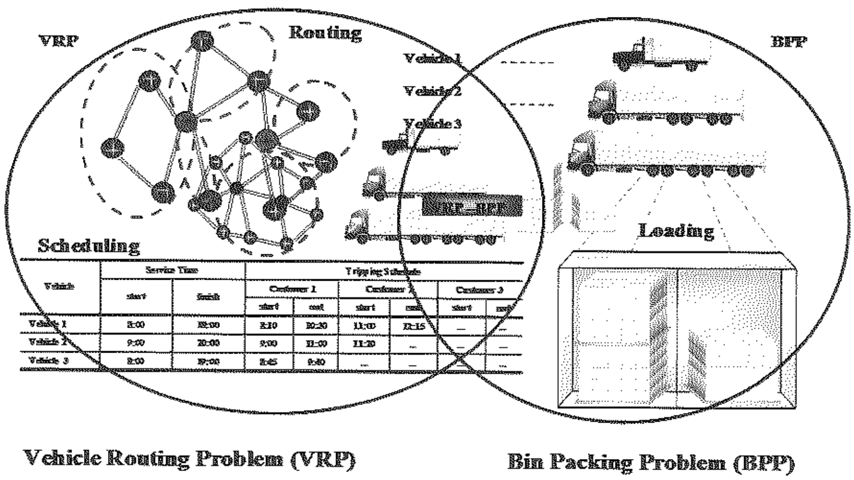

Fig. 1. Definition of the Routing-Scheduling-Loading Problem (RoSLoP)

The case of study contains the next elements:

- A set of *ORDERS* to be satisfied at the facilities of the customers, formed by boxes of products with different attributes such as weight, high, product type, supported weight, packing type and beverage type.
- A set of n customers with independent service schedules at a facility j [$start\_service_j$, $end\_service_j$] and a finite capacity of attention of the vehicles.
- A set of depots with independent schedules, which have the possibility to request goods to other depots.
- A fleet of vehicles with heterogeneous capacity $Vehicles_d$ to transport goods, with a service time $service\_time_v$ and a time for attention at the facilities of the customers. The attention time tm_{vj} depends on the capacity of the vehicle and the available people for docking and loading the vehicles.
- A set of roads represented by the edges of the graph. Each road has an assigned cost C_{ij}, each one with a threshold of allowed weight $MAXLoad_{vj}$ for a determined vehicle v that travels towards a facility j, and a travel time t_{ij} from facility i to j.

The objective of RoSLoP is to get a configuration that allows the satisfaction of the set of *ORDERS* at the set of the customer facilities, minimizing the number of vehicles used and the distance traveled. This new formulation includes a model with 12 variants of VRP: CVRP, VRPTW, VRPMTW, MDVRP, SDVRP, sdVRP, VRPM, HVRP, CCVRP, DDVRP, rdVRP and OVRP, described in section 2.1.

5 Formulation of RoSLoP

Input sets

C	Set of customer facilities or vertex of the associated graph
ORDERS	Set orders to be satisfied
D	Set of depots to satisfy all the customer demands
$Vehicles_d$	Set of available vehicles in a depot d
K	Set of all existent routes in a graph
K_d	Set of routes to be covered by a depot $d \in D$
$Pallets_v$	Set of containers of a vehicle v. Each container has an associated pair $(h_{palletij}, w_{palletij})$, which represents the high and weight of a container i when a customer j is visited by a vehicle v.
$ITEMS_j$	A set of units of *ORDERS* for a customer j.

Parameters

$Capacity_{vj}$	Capacity of a vehicle v to visit a customer j.
$service\_time_v$	Service time of vehicle v.
tm_{vj}	Maneuver time of a vehicle v at facility j, associated to vehicle docking and loading
$MAXLoad_{vj}$	Upper limit for the load to be assigned to vehicle v to visit facility j
$MAX_{Vehicles_j}$	Upper limit fot the number of vehicles attended simultaneously at a facility j.
$start\_service_j$	The time when a facility j starts its operation
$end\_service_j$	The time when a facility j ends its operation
c_{ij}	Transportation cost to travel from facility i to j
t_{ij}	Transportation time to travel from facility i to j

Real variables

$Load_{vj}$	Assigned load in a vehicle v to visit facility j.
$arrive_{ik}$	Arrival time of a vehicle to facility i using route k.
$left_{ik}$	Departure time of a vehicle from a facility i using route k.
φ_k	Associated cost to travel by route k.
t_k	Travel cost of a vehicle by route k.

Integer variables

x_{ijk}	1 if the edge (i, j) is visited on route k, 0 in otherwise
y_{vk}	1 if vehicle v is assigned to route k, 0 in otherwise

5.1 Preprocessing of the instance of RoSLoP

The preprocessing of an instance is carried out through a linear transformation function, which normalizes the load objects of the problem. Load objects are defined as n-dimensional objects. They are transformed into a set of real numbers where each number represents an n-dimensional object. The function of transformation is illustrated in Fig. 2, in which, an order of a customer j is transformed. This function represents the relationship between the dimensions of the objects. It consists of two steps: 1) the construction of units of load and 2) the transformation of these units in a representative set of real numbers.

$$f:\square^n \to \square$$

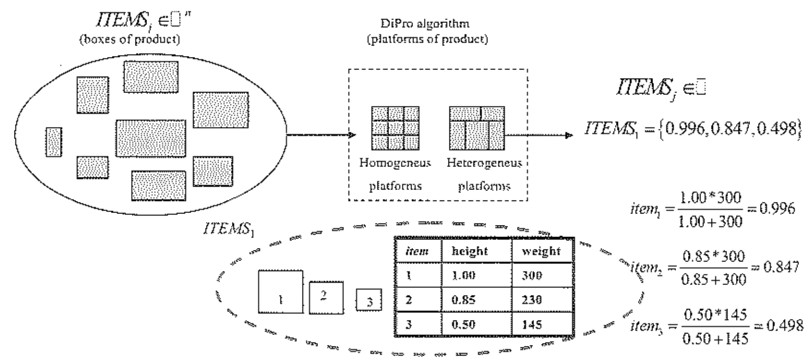

Fig 2. Transformation of the orders dataset of the case study

The construction of load units is done through the DiPro is invoked. As a result, two kinds of units are created: homogeneous and heterogeneous platforms. Homogeneous platforms are constituted by products of the same type, while heterogeneous platform are constituted with different types of products with similar characteristics. Both, Homogeneus and Heterogeneus platforms are defined as a set $ITEMS_j = \{\forall(w_i,h_i)\}$. Then, each pair (w_i, h_i) is transformed into a number $item_i$ using the following expression (1). A detailed review of DiPro is presented in [20].

$$item_i = \frac{h_i w_i}{h_i + w_i} \qquad i \in ITEMS_j \tag{1}$$

The capacity $Capacity_{vj}$ of a vehicle v to visit node j is transformed likewise. Each container that belongs to a trailer has two attributes: a high $hpallet_{ij}$ and weight $wpallet_{ij}$ of the assigned load to visit customer j. The width of the load is determined by a categorization of products, asked the company to group the products. This is necessary for adjusting the load to the containers. The transformation of the vehicles dimensions is shown in Fig. 3.

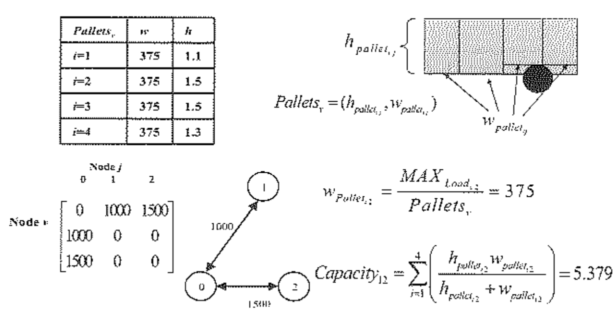

Fig. 3. Transformation of the vehicles dimensions

It is assumed that the weight of the load to be assigned to each container must be uniform. Expression (2) is used to obtain the capacity of the vehicle. This expression ensures that the dimensions of the load objects and the vehicles are equivalent.

$$Capacity_{vj} = \sum_{i=1}^{Pallets_v} \frac{h_{pallet_{ij}} w_{pallet_{ij}}}{h_{pallet_{ij}} + w_{pallet_{ij}}} \qquad j \in C, v \in Vehicles_d \qquad (2)$$

$$Load_{vj} = \sum_{r \in ITEMS_j} item_r \sum_{i \in C \cup D} x_{ijk} \qquad j \in C \cup D, k \in K_d \qquad (3)$$

Once defined the input parameters, instance preprocessing is performed. These elements are used to formulate the integer programming model. The combination of these elements generates the solution to the related rich VRP.

5.2 Mathematical model to solve the rich VRP

The objective of RoSLoP is to minimize the assigned vehicles and the distance traveled, visiting all the customer facilities and satisfying the demands. Each route k is constituted by a subset of facilities to be visited and a length φ_k. Expression (4) is used to get the maximum covering set established by the use of variant HVRP. Expressions (5)-(6) permit obtaining the length and the travel time on a route k.

$$|K_d| = |Vehicles_d| \left\lceil \frac{max(ITEMS_j)}{min(Capacity_{vj})} \right\rceil \qquad K_d \in K \qquad (4)$$

$$\varphi_k = \sum_{j \in C \cup D} \sum_{i \in C \cup D} c_{ij} x_{ijk} \qquad k \in K, v \in Vehicles_d \qquad (5)$$

$$t_k = \sum_{j \in C \cup D} \sum_{i \in C \cup D} t_{ij} x_{ijk} + \sum_{i \in C \cup D} \sum_{j \in C \cup D} \sum_{v \in Vehicles_d} tm_{vj} x_{ijk} y_{vk} \qquad k \in K_d \qquad (6)$$

The objective function of the problem, defined by expression (7), minimizes the number of assigned vehicles and the length of all the routes generated. Expressions (8) – (10) are used to generate feasible routes and solve the related TSP problem. Expression (8) restricts each edge (i, j) on a route k to be traversed only once. Expression (9) ensures that route k is continuous. Expression (10) is used to optimize the covering set related with the objective function. These expressions solve variants DDVRP and MDVRP.

$$\min \sum_{k \in K_d} \sum_{v \in Vehicles_d} \varphi_k y_{vk} \qquad (7)$$

$$\sum_{i \in C \cup D} x_{ijk} = 1 \qquad k \in K_d, j \in C \cup D \qquad (8)$$

$$\sum_{i \in C \cup D} x_{ijk} - \sum_{i \in C \cup D} x_{jik} = 0 \qquad k \in K_d, j \in C \cup D \qquad (9)$$

$$\sum_{i \in C \cup D} \sum_{j \in C \cup D} x_{ijk} \geq 1 \qquad k \in K_d \qquad (10)$$

Expressions (11)-(14), formulated in [5], calculate the time used by a vehicle assigned to route k. Expression (14) ensures that the use of a vehicle does not exceed the attention time at facility j.

These expressions permit solving the variants VRPTW, VRPMTW and VRPM. The variants CCVRP and SDVRP are solved using the expression (15), which ensures that two routes k and k' do not intersect each other at a facility j.

$$t_k y_{vk} \le service\_time_v \qquad k \in K_d; v \in Vehicles_d \qquad (11)$$

$$left_{jk} \ge arrive_{jk} \sum_{i \in C \cup D} t_{ij} x_{ijk} + \sum_{v \in Vehicles_d} tm_{vj} y_{vk} \qquad j \in C \cup D, k \in K_d \qquad (12)$$

$$arrive_{jk} \sum_{i \in C \cup D} x_{ijk} = \sum_{i \in C \cup D} t_{ij} x_{ijk} \qquad j \in C \cup D, k \in K_d \qquad (13)$$

$$start\_service_j \le arrive_{jk} \le end\_service_j \qquad j \in C, k \in K_d \qquad (14)$$

$$arrive_{jk} \le left_{jk} \le arrive_{jk'} \qquad k < k', \forall k, \forall k' \in K_d \qquad (15)$$

Expressions (16)-(18), formulated in [1] and combined with the linear transformation function, define the restrictions for variants CVRP, sdVRP, rdVRP and HVRP. Expression (16) establishes that a vehicle is assigned to a route k. Expression (17) ensures that vehicle capacities are not exceeded. Equation (18) establishes that all goods must be delivered and all demands are satisfied. The relaxation of the model that permits the solution of the variant OVRP consists of the reformulation of expression (9) through expression (19).

$$\sum_{v \in Vehicles} y_{vk} \le 1 \qquad k \in K_d \qquad (16)$$

$$Load_{vj} \le Capacity_{vj} y_{vk} \qquad k \in K_d; v \in Vehicles_d \qquad (17)$$

$$|ITEMS_j| - \sum_{j \in C} Load_{vj} = 0 \qquad j \in C \cup D \qquad (18)$$

$$\sum_{i \in C \cup D} x_{ijk} - \sum_{i \in C \cup D} x_{jik} \le 1 \qquad k \in K_d, j \in C \cup D \qquad (19)$$

The right side of expression (19) is 0 when a route starts and ends at a depot; otherwise, when the right side has the value 1 means that route starts at a depot and finishes at a different facility.

6 Experimentation

Real instances were provided by the bottling company. They were solved using the approximate algorithm called Heuristics-Based System for Assignment of Routes, Schedules and Loads (HBS-ARSL) proposed in [3] and an exact method. Both were tested on a set of VRP variants that are present in the test data set: VRPTW, VRPMTW, sdVRP, SDVRP, rdVRP, CCVRP, DDVRP, CVRP y HVRP. The HBS-ARSL algorithm was coded in C# and it was executed during two minutes to observe the time when it reaches the best solution. The implementation of the exact method was coded in C# and uses the LINDO API v4.1. A set of 12 test instances were selected from the database of the company, which contains 312 instances classified by the date of the orders; the database contains also 1257 orders and 356 products in its catalogues. Eight available vehicles were disposed. The results are shown in Table 3.

Table 3. Experimentation with real-world instances, provided by the bottling company

Instance	*n*	/*K*/	*ORDERS*	HBS-ARSL			Exact method		
				Distance Traveled	Vehicles Used	Time (secs)	Distance Traveled	Vehicles Used	Time (secs)
06/12/2005	**4**	5	158	**1444**	**4**	**51.33**	**1444**	**3**	**3.09**
09/12/2005	5	5	171	1580	5	23.64	1580	5	3.32
12/12/2005	7	9	250	2500	6	38.52	2500	6	5.03
01/01/2006	**6**	9	286	**2560**	7	**75.42**	**2560**	**6**	**3.42**
03/01/2006	4	4	116	1340	4	63.00	1340	4	2.97
07/02/2006	**6**	9	288	**2660**	7	**83.71**	**2660**	**6**	**4.53**
13/02/2006	5	7	208	1980	5	55.57	1980	5	3.85
06/03/2006	6	7	224	1960	5	32.16	1960	5	3.36
09/03/2006	6	9	269	2570	6	76.18	2570	6	3.85
22/04/2006	8	11	381	3358	7	57.35	3358	7	5.24
14/06/2006	7	8	245	2350	6	90.84	2350	6	4.86
04/07/2006	7	9	270	2640	6	72.49	2640	6	4.53
Average	**6**	**8**	**238.83**	**2245.16**	**5.66**	**55.27**	**2245.16**	**5.33**	**4.00**

7 Analysis of Results

The exact method obtained the optimal solution for the 12 instances of the test data set; which permits to measure the performance of the algorithm HBS-ARSL when solving the related rich VRP variant. Table 3 shows that HBS-ARSL reaches the optimal solution in 100% of the cases considering the distance traveled and in 75% considering the number of vehicles assigned. These results reveal as consequence, the need of improving the search techniques of HBS-ARLS to reach 100% of efficiency. The best solutions of HBS-ARLS were obtained in 55.27 seconds on average, while the exact method reaches the optimal solutions in 4 seconds, permitting a reduction of 92% in execution time; which reveals the advantages of the transformation function used to reduce the execution time and the computation of the optimal solution.

8 Conclusions and Future Works

This work presented a mathematical formulation and a linear transformation function, which make possible to obtain the optimal solution for the rich VRP variant related to RoSLoP. It was demonstrated that, the use of mathematical artifices can be used to reduce the dimensionality generated by the solution of many VRP variants. This allowed the obtaining an optimal solution through the formulated integer problem. It could be advantageous when an exact solution is needed for problems classified as NP-Hard as VRP or BPP. However, this transformation function has the restrictive condition of the dependence of domain for this application. Therefore, it is proposed the construction of a transformation function, which is able to reduce the dimension of some specific problems as BPP to a representative set. This can be used to obtain the optimal solution for other real-world problems.

References

1. Toth P. Vigo D, editors. The vehicle routing problem, SIAM Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics. (2001)
2. Cruz L. et al.: A Distributed Metaheuristic for Solving a Real-World Scheduling-Routing-Loading Problem, ISPA 2007, Ivan Stojmenovic (eds.), Springer Verlag, pp. 68-77. (2007)
3. Cruz L. et al.: An Ant Colony System to solve Routing Problems applied to the delivery of bottled products, ISMIS 2008, in Aijun An (eds.), Springer Verlag. pp. 329-338. (2008)
4. Shaw P.: Using Constraint Programming and Local Search Methods to Solve Vehicle Routing Problems," 40 Conference on Principles and Practice of Constraint Programming (CP 98), M. Maher and JF. Puget (eds.), Springer-Verlag, pp. 417-431. (1998)
5. Cordeau, F., Desaulniers, G., Desrosiers, J., Solomon, M., Soumis, F.: The VRP with time windows. Technical Report Cahiers du GERAD G-99-13, Ecole des Hautes ´Etudes Commerciales de Montreal (1999)
6. Mingozzi A.: An exact Algorithm for Period and Multi-Depot Vehicle Routing Problems. Department of Mathematics, University of Bologna, Bologna, Italy. (2003)
7. Archetti C., R. Mansini and M.G. Speranza.: The Vehicle Routing Problem with capacity 2 and 3, General Distances and Multiple Customer Visits. Operational Research in Land and Resources Manangement, p. 102. (2001)
8. Thangiah S.: A Site Dependent Vehicle Routing Problem with Complex Road Constraints. Artificial Intelligence and Robotics Laboratory, Slippery Rock University, U.S.A. (2003)
9. Dorronsoro B.: The VRP Web. AUREN. Language and Computation Sciences of the University of Malaga. http://neo.lcc.uma.es/radi-aeb/WebVRP. (2005)
10. Bianchi Leonora.: Notes on Dynamic Vehicle Routing. Technical Report IDSIA-05-01. IDSIA - Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Switzerland. (2000)
11. Jacobs Blescha and Mark Goetshalckx.: The Vehicle Routing Problem with Backhauls: Properties and Solution Algorithms. Techincal report MHRC-TR-88-13, Georgia Institute of Technology. (1993)
12. Fleischmann B.: The Vehicle routing problem with multiple use of vehicles. Working paper, Fachbereigh Wirtschaftswissenschaften, Universitt Hamburg. (1990)
13. Taillard, E.: A Heuristic Column Generation Method For the Heterogeneous Fleet VRP. Istituto Dalle Moli di Studi sull Inteligenza Artificiale, Switzerland. CRI-96-03. (1996)
14. Hasle G., O. Kloster, E. J. Nilssen, A. Riise, T. Flatberg.: Dynamic and Stochastic Vehicle Routing in Practice. Operations Research/Computer Science Interfaces Series, Springer, Vol. 38, pp 45-68, (2007).
15. Goel, A.; Gruhn, V. Solving a Dynamic Real-Life Vehicle Routing Problem. Haasis, H.-D.et al. (Ed.): Operations Research Proceedings 2005, Bremen, Deutschland. (2005)
16. D. Pisinger and S. Ropke, A General Heuristic for Vehicle Routing Problems, tech. report, Dept. of Computer Science, Univ. Copenhagen. (2005)
17. I. Cano, I. Litvinchev, R. Palacios and G. Naranjo.: Modeling Vehicle Routing in a Star-Case Transporation Network. XVI International Congress of Computation (CIC-IPN). (2005)
18. Herrera, J. Development of a methodology based on heuristics for the integral solution of routing, scheduling and loading problems on distribution and delivery processes of products. Master's Thesis. Posgrado en Ciencias de la Computación. Instituto Tecnológico de Ciudad Madero, México. (2006).
19. Rangel, N. Analysis of the routing, scheduling and loading problems in a Products Distributor. Master's Thesis. Posgrado en Ciencias de la Computación. Instituto Tecnológico de Ciudad Madero, México. (2005).
20. Cruz L. et al.: DiPro: An Algorithm for the Packing in Product Transportation Problems with Multiple Loading and Routing Variants, MICAI 2007, in Alexander Gelbukh and Angel Fernando Kuri Morales (eds.), Springer-Verlag, pp. 1078-1088. (2007)