© International Journal of Combinatorial Optimization Problems and Informatics, Vol. 9, No. 3, Sep-Dec 2018, pp. 55-63. ISSN: 2007-1558.

ILS Metaheuristic to solve the Periodic Vehicle Routing Problem

Tenahua A.[1], Olivares-Benítez E.[2], Sánchez-Partida Diana[3], Caballero-Morales S.O. [4]
[1,3,4] *Centro Interdisciplinario de Posgrados, UPAEP, Puebla*
[2] *Facultad de Ingeniería, Universidad Panamericana, Guadalajara*
[1,3,4] *angelicamaria.tenahua, diana.sanchez, santiagoomar.caballero@upaep.mx*
[2] *eolivaresb@up.edu.mx*

Abstract. This article presents a methodology for solving the Periodic Vehicle Routing Problem (PVRP) with an Iterated Local Search Metaheuristic (ILS). The problem is solved in two phases: the first step is to assign days of visit to each customer, and in the second step to determine the routes that each vehicle must perform each day. The heuristic for a local improvement in ILS is Clarke & Wright Heuristic, and perturbation is made on the days of visit assigned to some customers. The instances generated by Cordeau for PVRP with 51, 102 and 153 customers are used. The results are compared to the best-known solutions. The gap between the results presented by the proposed metaheuristic range from 15% to 5% above the best known solutions. The time to find the solutions with the proposed metaheuristic goes from 6.76 seconds for instances of 51customers, to 172.09 seconds for instances of 153 customers.

Keywords: Periodic Vehicle Routing Problem, Iterated Local Search, Two-Opt, Clarke & Wright Savings Algorithm.

1 Introduction

Within the logistics costs transport is the highest, and therefore there are several studies to reduce it. The main focus is on land transport, although it has also studied transport by water, air and even in space. [34] analyzed transport in space, in which the problem is to minimize the time, energy and economic costs of moving resources from one place to another in space. [32] analyzed the Oil Platform Transport Problem (OPTP), and conclude that is an NP-hard problem, which opens the door to the resolution of the water transport problem through Metaheuristics. The impact on the decrease in transport costs begins with a good strategy for locating the plants and facilities, as mentioned [33] which presents a proposal that allows establishing the relationships between the facilities location problem and the client allocation within a dense demand environment in territorial design. The location obtained means having available the decimal geographic coordinates in longitude and latitude from the location point in such a way that the products or services transfer has a minimum cost. The organization can design an efficient logistics plan to benefit its supply chain.

The Vehicle Routing Problem (VRP) is a widely studied problem of Combinatorial Optimization. The first publications about this problem were made around 1959 [15], where routes were generated at a minimum cost for a set of vehicles of homogeneous capacity [1]. Variations of the VRP have been studied including characteristics of real-world transport models, like: vehicles with heterogeneous fleets, time windows, periodic visits, more than one depot, etc. The main variants of VRP can be found in work presented by Toth and Vigo [31]. The PVRP can be thought of as a generalization of a conventional VRP, which seeks to determine an optimal set of daily routes for a given time horizon. Customers need to be visited on different days during the planning horizon, according to their demands, storage space, sales, etc. We call itineraries for the combinations of visits requires for each customer.

Solve the PVRP implies solve two entangled problems: assignment problem, and the vehicle routing problem. In the first, the decision is determinate a set of visiting days (itinerary) for each customer within the planning horizon. The other problem is the routing of vehicles for each day. The PVRP is an NP-Hard problem [13], [25], because includes the VRP with single period as a special case, and the most efficient solution techniques for these problems are metaheuristics. Among the most used metaheuristics for VRPs are Tabu Search [11], GRASP [28], Ant Colony Optimization [4], Variable Neighborhood Search [19] and Hybrid Heuristics based on Coverage of Sets [5].

The total cost to be minimized may include the costs associated with the distances travelled, vehicle capacity, transit time, fuel, etc. For this work, the distance traveled is directly related to the cost. To have a detailed discussion about costs, the reader can

Received Dec 26, 2016 / Accepted Dec 11, 2017

check the publication [14] in which a review of fixed and variable transport costs is made. Several real-world applications deal with transport (delivery trucks) of the same capacity, known as homogeneous fleet. In this work we considered a planning horizon of 6 days, a week from Monday to Saturday, the most common application. The application of PVRP is wide, some examples are in recyclable and organic garbage collection, laundry service to hotels, the gas services to retailers, etc. that is why we consider a horizon of six days. To find the PVRP solution the assignment of itinerary will be made to each customer according to the number of visits that are required during the planning horizon. Once the customers to be visited each day are known, routing is performed using the Savings Heuristic [8].

2 Literature Review

Gaudioso and Paletta [18] described a model for the optimum administration of periodic deliveries for a given product. The objective was to minimize the planning horizon of the maximum number of vehicles used simultaneously (i.e. the size of the fleet). They achieve a balanced use of resources for the class of instances solved. Francis & Smilowitz [17] presented a continuous approach model for PVRP with service choice. The results obtained can help distribution service providers to design value-for-service options.

One of the most used tools to solve routing problems is the heuristic proposed by Clarke & Wright [8]. Ballou & Agarwal [3] made a comparison between the Savings, Cluster and Sweeping Methods, under five types of population distribution: Random, Grouping, Sector, Urban-Rural and Coastal. The results reflected an advantage of the Savings Method in the different types of distribution. They made a comparison between results obtained from five different instances of the type of population distribution, through ACO metaheuristics, and Clarke & Wright Heuristics [8]. Among the results obtained, the best were those of the Sector type. Clarke & Wright Heuristics generate a good solution for the VRP, and that is why we used this heuristic in this work.

Pacheco et al. [28] used PVRP to give a solution to the routes of a bakery of the north of Spain. They applied GRASP and Path Relinking and took the instances of Cordeau et al. [26] to compare their solutions. The solutions were updated like best-known solutions. We decided to use ILS Metaheuristic, described below, to compare the effectiveness.

ILS is simple in the algorithm mechanisms but powerful in its scope. ILS has been applied to many combinatorial optimization problems with great success, including permutation flow problems [16] and CVRP [6]. For more details, see [23]. Iterated Local Search (ILS) is a metaheuristic which proposes a scheme in which a basic heuristic is included to improve the results of the repetition of that heuristic. ILS given a solution obtained by the application of a basic heuristic, a change or alteration is made that gives rise to an intermediate solution. The application of basic heuristic to this new intermediate solution provides a new solution that, if it passes an acceptance test, becomes the new perturbed solution. Although the basic heuristic included, usually is a local search, it has been proposed to apply any other heuristic, deterministic or not. In this way, the process becomes a stochastic search for environments, where such environments are not explicit, but are determined by the basic heuristic [24].

Cacchiani et al. [5] used ILS to generate columns for the LP relaxation. They perform a benchmark with the instances of [28] to show the effectiveness of the proposed algorithm for the generation of good quality solutions. Abreu et al. [1] used ILS to solve the PVRP. To improve the disturbed solution they used four structures between routes and three within routes. The changes between routes were Shift (h,h), Swap (h,h), Cross and Radial. The changes within a route were by Swap (1), 2-Opt, Or-Opt (1). They compared the results obtained by CPLEX and ILS. The solutions by ILS were equal or better than those of CPLEX with runs with a time limit of one hour. The instances compared was from 12 to 30 customers.

Archetti et al. [9] introduced a new problem called Flexible Periodic Vehicle Routing Problem (FPVRP), related to the Inventory Routing Problem (IRP), and is a new and challenging problem dealing with flexibility in periodic delivery operations. They used three different sets of benchmark instances of relatively small size. The results show that FPRVP may produce improvements in the routing costs in comparison with both PVRP and IRP.

Iterative improvement methods modify a current solution through local searches to get better neighbourhoods of the solution. In general a neighbourhood comprises a set of solutions that can be obtained by changing a subset r of arcs between the solutions. An exchange of arcs is performed only if that change leads to an improved feasible solution. This exchange can be done in or between routes. The process ends when a solution is found with r-optimum, or if could not be improved by more than r-exchanges. The first improvement procedures were proposed in 1977 by [30], [10] and [2]. Although these authors maintain a small r, $r = 2$ or 3, the neighbourhoods generated are very large. This leads to efficient but laborious methods [31]. Most improvement procedures for TSP (Traveling Salesman Problem), where PVRP come from, can be described in terms of Lin's X-opt mechanism [21]. [22] modified A dynamically throughout the search. [27] proposed the O-opt method, moving 3, 2, or 1 consecutive vertex to another location. [26] developed a restricted version of the 4-opt algorithm, called 4-opt*. [20] carried out a thorough empirical analysis of these and other improvement procedures for the TSP, concluding that a careful application of the schemes yields the best results in average [22].

The previous literature review helps us to distinguish the differences of our work with the works of other authors. It was based on the mathematical model proposed by [17]. Demands and visit requirements are fixed. There is no restriction on the size of the fleet. We use ILS metaheuristics to obtain the PVRP solution. Initially, the first improvements within the routes were made under Two-opt method, which as mentioned previously generates good solutions. However, Clarke & Wright Heuristic improved the results that are why is considered for the improvement of routes. We solved the instances [11] to know the efficiency of our results, benchmarking Cordeau instances with respect to the total distance travelled and time. The results are very close. In this work, the instances are greater than those presented by Abreu et al. [1]. The smallest instance in this work is 51 customers, and the largest is 153 customers.

3 Problem description

Reports Consider the graph **G**=(**N,A**) and planning horizon of ***T*** days (periods, *t*), where ***N*** is the set of nodes representing customers, and ***A*** the set of arcs connecting the nodes. Each customer requires the collection task with a frequency f_i, measuring the number of times the customer *i* must be visited with $1\leq f_i\leq T$. The basic PVRP consist in selecting f_i visits days for node *i* and solve *np* vehicle routing problems, one for each period of planning horizon in order to minimize the total cost travel.
Was proposed the PVRP model as follows [7]:

- S_i the set of feasible itineraries for customer *i*;
- x_{ik} = 1 if the *k-th* itinerary is selected for the customer *i*, 0 otherwise;
- *n* is the total number of customers and *T* the number of days in the period;
- $a_{kt}=1$ if day *t* is on the itinerary *k*, 0 otherwise;
- q_i is the demand of the customer *i* for each delivery
- c_{ij} is the distance from customer *i* to customer *j*;
- $N = (i|i=1,2,\ldots,n)$ is the set of customers;
- Q_r is the vehicle capacity *r* and D_r is the allowed driving time for the vehicle *r*;
- R_t is the given Set of available vehicles for the day *t*;
- $V_{it} = 1$ if the customer *i* is visited on day *t*, 0 otherwise. The depot is represented by customer *0* such that $V_{ot} = 1,\ t=1,\ldots,T$; and be
- $u_{ijtr} = 1$ if vehicle r R_t goes from *i* to *j*, on day *t*, 0 otherwise.

With these variables the problem was defined as follows:

Minimize

$$\sum_{t=1}^{T}\sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{r\in R_t} c_{ij}u_{ijtr} \tag{1}$$

Such that

$$\sum_{k\in S_i} x_{ik}=1, \forall\ (i\neq 0) \tag{2}$$

$$v_{it}=\sum_{k\in S_i} x_{ik}a_{kt}, \forall\ t, \forall\ i(i\neq 0) \tag{3}$$

$$\sum_{r\in R_t} u_{ijtr}\leq \frac{v_{it}+v_{jt}}{2}, \forall\ i,j,t(i\neq j), \tag{4}$$

$$\sum_{i=0}^{n} u_{iptr}=\sum_{j=0}^{n} u_{pjtr}, \forall\ p,t,r\in R_t, \tag{5}$$

$$\sum_{r\in R_t}\sum_{i=0}^{n} u_{ijtr} = v_{jt}, \quad \forall j,t (j \neq 0)$$
$$= |R_t| \quad \forall t (j=0) \tag{6}$$

$$\sum_{i\in W}\sum_{j\in W} u_{ijtr} \leq |W| - 1, \quad \forall t, r \in R_t, \forall W \subseteq N, \tag{7}$$

$$\sum_{j=1}^{n} u_{0jtr} \leq 1, \quad \forall t, r \in R_t, \tag{8}$$

$$\sum_{i=1}^{n} q_i \left(\sum_{j=0}^{n} u_{ijtr} \right) < Q_r, \quad \forall t, r \in R_t, \tag{9}$$

$$x_{jk} \in (0,1), \quad \forall i,k \in S_i, \tag{10}$$

$$u_{ijtr} \in (0,1), \quad \forall i,j,t,r \in R_t, \tag{11}$$

Equation (1) is the Objective Function to evaluate the total cost of travel. (2) ensures that only one itinerary is selected for each customer. (3) ensures that a customer is only visited on a particular day if the itinerary chosen has a delivery on that day. (4) ensures that no vehicle can go between two customers on a particular day unless they both are scheduled for delivery on that day. (5) ensures that if a vehicle visits a customer, this vehicle leaves that customer. (6) ensures that each customer is visited on the days when the delivery was scheduled. (7) is the set of subtour elimination constraints, (8) ensures that a vehicle can only be used at maximum once. The constraints in (9) correspond to vehicle capacity. Finally, equations (10) and (11) indicate the domain of binary variables.

4 Methodology

ILS metaheuristics are simple in algorithm mechanisms, but powerful in their scope, that is why is used in this work for the PVRP. An initial solution (called *So*) is required, in this work is generated by random assignment of itineraries to customers, as they need to be visited. The routing for *So* is done in the sequence of assignment of the customers. The second ILS process requires an improvement in the initial solution (called *S*); done by the Clarke & Wright heuristic. The third ILS process indicates that once a better solution is obtained, it has to be altered. In this work, the alteration proposed is to take a percentage of the customers and change their itinerary (maintaining the requirements of visit frequency) giving rise to ILS intermediate solution (S'). Once the alteration is performed, is necessary generate the new routes. The new routes are generated again in order of assignment and improved by Clarke & Wright heuristic, thereby obtaining the improved intermediate solution (S").
The Solutions *S* and *S"* are compared by means of an acceptance test; the accepted solution is the new initial solution (*So*), with which the same process will be repeated n times. The test of acceptance in this work is one that gives a better solution. The solution is evaluated by the total distance travelled, such that the new *S* accepted is the one with the shorter distance, used to repeat the process.
ILS proposes the repetition of this process to explore neighbourhoods and to leave local minimums. The cycle described above is performed with a fixed number of repetitions. Further exploration of the neighbourhood (number of iterations and percentage of perturbation) is expected to find the best solution known. **Figure 1** describes the methodology.

1:	Random assignment of itineraries to customers
2:	Identification of nodes to be routed on each day of the period (1 to 6).
3:	Generate Initial solution *So:* Routes for each day, generated in the order of customer number.

4:	Generate *S* (Local Search): Each Route is enhanced by the Clarke & Wright Heuristic.
5:	Repeat *n* times Repeat Generate *S'*: The allocation of itineraries to a random *p*% of customers is modified. Generate *S''*: The routes for each day are obtained by the Clarke & Wright heuristic (local search). New *S*: Test of acceptance. Considering *S* and *S* ', the new *S* will be the one with the shortest total distance travelled. Update *S* as the best solution.
6:	Returns the best solution.

Figure 1. ILS Methodology

5 Experiments

According to Methodology presented, below are the details of the application. Is considered a planning horizon of 6 days, so *T*= *6*. Each customer *i*, has a visit frequency f_i associated. Is possible to represent the itineraries using **Table 1**. If a customer is visited on day *t*, the value is 1, 0 otherwise.

The instances generated by Cordeau et al. [11] are taken from the website: http://neo.lcc.uma.es/vrp/vrp-instances/periodic-vrp-instances/. The solutions are available, which allows comparing the results obtained by the Methodology presented.

There are 38 available Cordeau instances are for PVRP, however, for our study those that had 6-day itineraries were selected, that is 9 of the 38 instances. The reason was mention in the introduction section, according to the characteristics of services that need to be in a week. The instances are three of each size: 51, 102 and 153 customers. The details of the instances are available on the website: http://neo.lcc.uma.es/vrp/vrp-instances/description-for-files-of-cordeaus-instances/.

Table 1, presents the different itineraries according to the times that it is necessary to visit the customer, according to instances generated by Cordeau et al. [11]. For one visit there are 6 itineraries, for two visits 3 itineraries and one for daily visits. The itinerary assignment is random according to feasibility defined by f_i for each customer *i*.

Table 1. Itineraries for one, two or six visits in the week.

Number of Visits (fi)	Day 1	Day 2	Day 3	Day 4	Da y5	Day 6
6	**1**	**1**	**1**	**1**	**1**	**1**
2	0	0	1	0	0	**1**
2	0	1	0	0	**1**	0
2	1	0	0	**1**	0	0
1	0	0	0	0	0	**1**
1	0	0	0	0	**1**	0
1	0	0	0	**1**	0	0
1	0	0	**1**	0	0	0
1	0	**1**	0	0	0	0
1	**1**	0	0	0	0	0

In **Table 2** there is an example of routes generated each day, for a planning horizon of 6 days. The visit regime is represented by a set *P* of binary vectors *x*. Each component x_t takes the value 1 if that customer is visited on day *t*, 0 otherwise. For example, look at customers 4, 14 and 9. Customers need to be visited 1, 2 and 6 times a week respectively. For customer 4, the set *P* is *P* = {100000, 010000, 001000, 000100, 000010, **000001**} (one-day visit itineraries). The element of a set that was assigned is the last, so it is visited on day 6 of the planning horizon. For customer 14, the set *P* is *P* = {**001001**, 010010, 100100} (two-days visits itineraries) the element assigned is 1, so customer 14 is visited on day 3 and 6 of the period. Finally, the set *P* for customer 9 is only *P* = {**111111**} (six-days visits itinerary), and its assignment is the only element of the set *P*, whereby customer 9 is visited every day. A route begins and finish in 0. Day 1 has two routes, begin in 0, visit customer 9, 15, 10 and return 0, and begin the second route, that includes 3 and 12 customer.

Table 2. Example of routes generated each day for planning horizon.

Day	ROUTES										
1	0	**9**	15	10	0	3	12	0	0	0	0
2	0	2	6	5	0	8	12	**9**	0	0	0
3	0	15	6	<u>**14**</u>	0	**9**	0	0	0	0	0
4	0	15	1	5	0	**9**	11	0	3	12	0
5	0	8	**9**	2	0	3	13	0	0	0	0
6	0	6	7	**9**	0	***4***	11	<u>**14**</u>	0	0	0

For the metaheuristic detailed in the previous paragraphs, a program in C ++ language was developed, and 9 instances are considered. Basic information of instances is presented in Table 3. The norm distance to generate the matrices needed for routing process is Euclidian.

Table 3. Cordeau Instances Data, studied in this paper.

Instance name	Numbers of Nodes	Numbers of days of the period	Maximum capacity of vehicles	total Itineraries
p24, p25, p26	51	6	20	10
p27, p28, p29	102			
p30, p31, p32	153			

Perturbations to *S'* to explore the neighbourhood, was made to 10, 20 and 30% of customers. This means that the visit days of selected customers (10, 20 and 30%) changed with a different itinerary. The best results were obtained with the 30% alteration in the assignment itineraries to customers. At the begining 100 replicates were made, after 300, later 500, and finally 2,500 replicates, observing that as the number replicates increases, the solution improves. **Table 4** shows the results obtained for instance p28, with 102 customers. As can observe, the best solutions are obtained with 30% of perturbation that is why perform the 30% perturbation was made with 2500 replicates.

Table 4. Results for Instance p28 with 102 customers.

Replicates	% Alteration	Best Solution Known	Average solution	Average Time	% Difference	Best solution ILS	Time	% Difference
100	10	22934.71	25607.40	6.78	**11.80**	25339.70	6.76	10.49
	20		25757.26	6.81	12.31	25415.17	6.79	10.82
	30		25752.57	6.82	12.29	25008.63	6.82	9.04
300	10		25508.51	20.59	11.22	25254.97	20.58	10.12
	20		25474.02	20.61	11.07	25247.11	20.65	10.08
	30		25459.23	20.64	**11.01**	24582.34	20.41	7.18
500	10		25447.65	34.35	10.96	25013.22	34.51	9.06
	20		25499.75	34.16	11.18	25203.58	34.40	9.89
	30		25367.24	34.38	**10.61**	25232.28	34.37	10.02
2500	10		25164.98	172.03	**9.72**	24861.37	171.21	8.40
	20		25246.68	172.44	10.08	24767.56	172.09	7.99
	30		25169.01	171.55	9.74	**24561.64**	171.21	**7.09**

The execution times for 51 nodes instances came from an average of 1.328 seconds for 100 replicates, 3.893 seconds for 300 replicates, to 6.41 seconds for 500 replicates. This tune the implementation for making 2,500 replicates, since although the average time was 32.76559121 seconds, the solution improved by 6%. With 100 replicates the solution was about 20% far from the best solution known, and went far to 14.1068% with 2,500 replicates. This indicates that as the number of replicates increases the solution improves. All solutions improved with the increase of the replicates and 30% of perturbation assignment of customers to itineraries.

The results of interest are, total distance travelled and execution time, given in seconds. In **Table 5**, we present the average results of instances 24, 25 and 26, when executing 10 times the metaheuristic, indicating in bold the best solution found with ILSThe percentage of difference indicates how much the solution found by ILS, is over the best published. It also presents the execution time and what percentage of the best solution was the best solution found by ILS with 10 replicates. In **Table 6**, present the results of instances 27, 28 and 29, whose percent difference decreases even as the number of customers increases. Finally, the results of instances 30, 31 and 32 are presented in **Table 7**.

Table 5. Average results of 10 replicates with instances of 51 nodes.

Instance	Best Solution Found [23]	Average Solution	Average Execution Time (seconds)	% Difference	Best solution ILS	Time for the best solution ILS	% Difference
p24	3687.46	4219.93	32.76	14.44	**4113.50**	**33.01**	**11.55**
p25	3777.15	4337.60	32.75	14.84	**4314.71**	**33.07**	**14.23**
p26	3795.33	4374.27	32.23	15.47	**4349.15**	**32.26**	**14.59**

Table 6. Average results of 10 replicates with instances of 102 nodes.

Instance	Best Solution Found [23]	Average of Solutions	Average Execution Time (seconds)	% Difference	Best Solution ILS	Time for the best solution ILS (seconds)	% Difference
p27	21956.46	24863.06	171.40	13.24	**24431.39**	**171.68**	**11.27**
p28	22934.71	25169.01	171.55	9.74	**24561.64**	**171.21**	**7.09**
p29	22909.36	25835.48	170.01	12.77	**25500.38**	**168.84**	**11.31**

Note in **Table 7** that for instance p30 a solution is obtained with the ILS metaheuristic close to the best-published solution. Only 4.56% far from the best solution known. The advantage of the metaheuristic presented in this work is the time, since obtaining the solution took 10 minutes with 12 seconds, which is good for an instance of 153 nodes. In the case of instances with 51 nodes, the response time for the solution is 2.84 minutes.

LINGO was used to solve an instance of 10 customers, and the optimal solution was found in 53 seconds. However, for instance of 20 customers, the process was stopped after 405 hours, 59 minutes and 34 seconds, without obtaining an optimal solution.

Table 7. Average results of 10 replicate with instances of 153 nodes.

Instance	Best Solution Found [23]	Average of Solutions	Average Time	% Difference	Best Solution ILS	Time for the best solution ILS	% Difference
p30	80479.20	85369.83	605.39	6.08	**84151.36**	**607.53**	**4.56**
p31	78179.89	86160.22	604.38	10.21	**81780.65**	**606.51**	**4.61**
p32	75016.58	86332.24	605.96	15.08	**84110.73**	**607.53**	**4.51**

As the number of nodes increases, obviously the time to obtain the solution also does, but not exponentially. It is interesting to note also that as the size of the instance grows the solution that is obtained is close to the best solution found. **Table 8** shows the average execution time for the different instances and the percentage difference in distance with respect to the best solution found.

Table 8. Averages of times in seconds of execution and the difference in distance with respect to the best solution found for the different sizes of instances

Instance Size	Average Execution Time (seconds)	% Difference
51	32.58	13.46
102	170.99	9.89
153	605.24	4.56

6 Conclusions

Iterated Local Search is a metaheuristic that allows us to find solutions that are close to optimal in a reasonable time. This work shows that, as the size of the instance increases, the solution approaches the best one found. It is also important to mention that, likewise increasing the size of the instance increases the time to obtain the solution, but remains reasonable. In the same way as the percentage of perturbation for the ILS and the number of replicates increased, the solutions improved. It was decided to make use of the Cordeau instances, to be able to perform a benchmark with the solutions found and thus be able to know the quality of the solutions found by the proposed heuristics. A proposal for future work and improvement of the solutions is to limit the number of routes generated per day, hoping that the route balance will allow having a smaller fleet.
Another proposal is that, once you have customers assigned to each day, and if there are few, you could obtain VRP optimally for each day. This instead of using Clarke & Wright's heuristics with a Metaheuristic approach.
One more proposal takes instances from a company to compare the results obtained with ILS.

7 Acknowledgement

This research has been partially funded by Universidad Panamericana through the grant "Fomento a la Investigación UP 2017", under project code UP-CI-2017-ING-GDL-07

8 References

1. Abreu R., y Arroyo J.: Aplicação da heurística ILS para o Problema de Roteamento de Veículos Periódico com Frota Heterogênea e Custos Fixos. Simposio Latinoamericano de Investigación de Operaciones e Inteligencia Artificial pp. 1-9 (2015).
2. Baker E. and Schaffer J.: Computational experience with branch exchange heuristics for vehicle routing problems with time window constraints. American Journal of Mathematical and Management Sciences, vol. 6, pp. 261-300 (1986).
3. Ballou R., Agarwal Y.: A performance comparison of several popular algorithms for vehicle routing and scheduling. Journal of Business Logistics, Vol. 9, No.1, 1998, pp. 51-65.
4. Bell J. and Griffis S.: Swarm intelligence: Application of the ant colony optimization algorithm to logistics-oriented vehicle routing problems. Journal of Business Logistics, Vol. 31, No.2, pp.157-175 (2010).
5. Cacchiani V., Hemmelmayr V. and Tricoire F.: A set-covering based heuristic algorithm for a periodic vehicle routing problem, Discrete Applied Mathematics, Vol. 163, Parte 1, pp. 53-64 (2014).
6. Chen, P., Huang, H.K. and Dong, X.Y.: Iterated Variable Neighborhood Descent Algorithm for the Capacitated Vehicle Routing Problem. Expert Systems with Applications 37, pp. 1620–1627 (2010).
7. Christofides N. and Beasley J.E.: The Period Routing Problem, Networks, Vol. 14, Issue 2, pp. 237 – 256 (1984)
8. Clarke G. and Wright J.: Scheduling of Vehicles from a Central Depot to a Number of Delivery Points, Operations Research, Vol.12, Issue 4, pp. 568-581 (1964)
9. Claudia Archetti, Elena Fernández, Diana L. Huerta-Muñoz, The Flexible Periodic Vehicle Routing Problem, Computers and Operations Research (2017), doi: 10.1016/j.cor.2017.03.008
10. Cook T. and Russell R.A.: A simulation and statistical analysis of stochastic vehicle routing with timing constraints. Decision Science, 9, pp. 673-687 (1978).
11. Cordeau J., Gendreau M. and Laporte G.: A tabu search heuristic for period and multi-depot vehicle routing problems, Networks, Vol. 30, pp. 105-119 (1997)
12. Cordeau J, Laporte G and Mercier A.: A unified tabu search heuristic for vehicle routing problems with time windows, Journal of the Operational research society, Vol. 52, No.8, pp. 928-936 (2001)

13. Chao I., Golden B. and Wasil E.: An improved heuristic for the period vehicle routing problem, Networks, Vol. 26, Issue 1, pp.25-44 (1995)
14. Da Silva I., Reis R. and Gomes M.: Custos e otimização de rotas no transporte de leite a latão e a granel: um estudo de caso, Organizações Rurais & Agroindustriais, Vol. 2, No. 1 (2011)
15. Dantzig G. and Ramser J.: The Truck dispatching problem, Management Science, Vol. 6, No.1, pp. 80-91 (1959)
16. Dong, X.Y., Huang, H.K., Chen, P.: An Iterated Local Search Algorithm for the Permutation Flow-shop Problem with Total Flowtime Criterion. Computers & Operations Research 36, pp. 1664–1669 (2009).
17. Francis P. and Smilowitz K.: Modeling Techniques for Periodic Vehicle Routing Problems, Vol. 40, Issue 10, pp. 872-884 (2006)
18. Gaudioso M. and Paletta G.: A Heuristic for the Periodic Vehicle Routing Problem, Transportation Science Vol. 26, No.2, pp. 86-92 (1992).
19. Hemmelmayr V, Doerner K. and Harlt R.: A variable neighborhood search heuristic for periodic routing problems, European Journal of Operational Research, Vol. 195, No.3, pp. 791-802 (2009).
20. Johnson D.S. and McGeoch L.A.: The traveling salesman problem: A case study. In E.H.L. Aarts and J.K. Lenstra, editors, Local Search in Combinatorial Optimization, Wiley, Chichester, UK, 1997, pp. 215-310.
21. Lin S.: Computer solutions of the traveling salesman problem. Bell System Technical Journal, vol. 44, pp. 2245-2269, (1965).
22. Lin S. and Kernighan B.W.: An effective heuristic algorithm for the traveling salesman problem. Operations Research, vol 21, pp. 498-516 (1973).
23. Lourenco H.R., Martin O. and Stützle T.: Iterated local search. Cap. 11 en F. Glover y G.G. Kochenberger (eds.) Handbook of Metaheuristics , Kluwer Academic Publishers pp. 321–354, Boston (2003).
24. Melian B., Moreno J., Moreno M.: Metaheuristics: A global view. Departamento de Estadística, I.O. y Computación, Centro Superior de Informática Universidad de La Laguna Santa Cruz de Tenerife, Spain Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial. No.19, pp. 7-28 (2003).
25. Méndez A., Pontin M., Ziletti M. and Chávez L.: Heurísticas para la resolución de un Problema de Ruteo de Vehículos Periódico, Mecánica Computacional Vol. XXIV, pp. 2951-2960 (2005).
26. NEO, 2013, "Networking and emerging optimization: VRP instances", avaliable at: neo.lcc.uma.es/vrp/ (I., 1976)
27. Or I.: Traveling salesman-type combinatorial optimization problems and their relation to the logistics of regional blood banking. Ph.D. dissertation, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL (1976).
28. Pacheco J., Álvarez A., García I. and Angel-Bello F.: Optimizing vehicle routes in a bakery company allowing flexibility in delivery dates, Journal of the Operational Research Society, Vol.63, No.5, pp. 569-581 (2012)
29. Renaud J., Boctor F.F. and Laporte G.: A fast composite heuristic for the symmetric traveling salesman problem. INFORMS Journal on Computing, 8, pp.134-143 (1996).
30. Russell R.A.: An effective heuristic for the m-tour traveling salesman problem with some side conditions. Operations Research, vol, 25, pp. 517-524 (1977).
31. Toth P. and Vigo D. (2002). The Vehicle Routing Problem. Bologna, Italia. Society for Industrial and Applied Mathematics.
32. Díaz-Parra, O., Vanoye, J. A. R., Fuentes-Penna, A., Loranca, B. B., Pérez-Ortega, J., Barrera-Cámara, R. A., ... & Pérez-Olguin, N. B. (2017). Oil Platform Transport Problem (OPTP) is NP-hard. International Journal of Combinatorial Optimization Problems and Informatics, 8(3), 2-19.
33. Loranca, M. B., Velázquez, R. G., Benítez, E. O., & Flores, J. L. (2016). A Location Allocation Model for a Territorial Design Problem with Dense Demand. International Journal of Applied Logistics (IJAL), 6(1), 1-14. doi:10.4018/IJAL.2016010101
34. Ruiz Vanoye, J., Cedillo-Campos, M., Fuentes-Penna, A., & Díaz-Parra, O. (2016). Editorial for Volume 7 Number 1: The Space Transportation. International Journal of Combinatorial Optimization Problems and Informatics, 7(1), 1-2. Retrieved from https://ijcopi.org/index.php/ojs/article/view/40