Hybrid Algorithm for Solving the Multi-compartment Vehicle Routing Problem with Time Windows and Profit

Hadhami Kaabi[1] and Khaled Jabeur[2]
[1]Département Informatique, Institut Supérieur de Gestion, 41, Avenue de la Liberté, 2000, Tunis, Tunisia
[2]Département Méthodes Quantitatives, Institut Supérieur de Commerce et de Comptabilité, Rue Sadok El Jaouani, 7021, Tunis, Tunisia

Keywords: Multi-compartment, Vehicle Routing, Time Windows, Profit, Genetic Algorithm, Iterated Local Search.

Abstract: This paper presents a new variant of the well-known vehicle routing problem with time windows (VRPTW). More precisely, this paper addresses a multi-compartment vehicle routing problem with time windows and profit (MCVRPTW with profit). The aim of this problem is to serve a set of customers by using a set of vehicles with multiple compartments, under a minimum traveling cost. The vehicles, starting and ending at the depot, have a limited capacity and each compartment is dedicated to one product. A customer is served only within a given time windows and, when it is visited a profit is collected (i.e. a profit not low than a preset profit bound). To solve this problem, an hybrid approach combining the genetic algorithm (GA) and the iterated local search (ILS) is used.

1 INTRODUCTION

Nowadays, the efficient distribution of goods in any company enhances its competitiveness. To face the increase in energy costs, an efficient planning of the delivery paths should be established. This planning should save resources and money. A well known and studied problem to deal with this situation is the Vehicle Routing Problem (VRP). The VRP, introduced by Dantzig and Ramser (1959), is considered as a combinatorial optimization problem which is crucial for a huge area of applications (e.g. logistics, production, distribution, ect). The VRP is an extension of the classical Traveling Salesman Problem in which a set of customers are served with a fleet of vehicles based at one depot. In any classic VRP, the aim is to find a minimum cost routes such that: (i) each vehicle starts and ends at the depot, (ii) the total demand on each route should not exceed the vehicle capacity. Several variants of the VRP have appeared and were the subjects of intense studies such as: VRP with Pickup and Delivery, Periodic VRP, Capacitated VRP, VRP with time windows, Stochastic VRP, ect. The multi-compartment VRP is also a variant of the VRP but it has not received much attention yet and it is the topic of our research work.

The multi-compartment VRP (MCVRP) is introduced in 1979 and had practical significance (Reed et al., 2014). The MCVRP consists in transporting different products that should be kept in separate compartments because of incompatibility constraints. Each compartment is dedicated to one product. The demand of each customer for a product is delivered by one vehicle. However, the entire delivery (e.g. demand) of a customer can be brought by several vehicles (Fallahi et al., 2008). The MCVRP is NP-hard since it is a particular case of the VRP. Vehicle routing problems with compartments are well known in logistics but have little attention in the literature. The most published papers of the MCVRP concern these applications:

The distribution of various types of Fuel and oil ((Lahyani et al., 2015), (Relvas et al., 2014), (Cornillier et al., 2012), (Cornillier et al., 2009), (Cornillier et al., 2008), (Fagerholt et al., 2000)) and some maritime applications ((Stalhane et al.,2012), (Havattum et al., 2009)). A second application involves the delivery of animal foods introduced by Fallahi et al. (2008), and grocery products requiring different levels of refrigeration (e.g. dry, refrigerated and frozen) (Chajakis and Guignard, 2003). The pickup and delivery of livestock is also a known application of the MCVRP as developed in (Oppen and Lokketangen, 2008). Animals should be kept separately to avoid contamination. Multi-compartment vehicles are also used in waste collection, where for example the general waste

Kaabi H. and Jabeur K..
Hybrid Algorithm for Solving the Multi-compartment Vehicle Routing Problem with Time Windows and Profit.
DOI: 10.5220/0005572503240329
In *Proceedings of the 12th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2015), pages 324-329
ISBN: 978-989-758-122-9

are loaded in a specified compartment and another compartment is dedicated to recyclables ((Reed et al., 2014), (Muyldermans and Pang, 2010)).

The MCVRP with time widows (MCVRPTW) and profit extends the MCVRP. First, each customer is served only within a given time windows, which is a common constraint in various routing problem (e.g. vehicle routing problem with time windows (VRPTW)) (Solomon, 1987). Second, the request of a customer is composed of different products with which profits are associated . Once a request is satisfied , a profit is collected. A customer is visited if and only if a preset lower bound on the profit is exceeded. This constraint implies that some customer requests may not be satisfied. This problem is addressed under different names in the literature. A traveling Salesman Problems with profits was provided by Feillet et al. (2011), ((Aras et al., 2011), (Valle et al., 2011)) have also proposed a selective vehicle routing problem. In this paper, we propose an hybrid algorithm to solve the MCVRP with time windows and profit. To the best of our knowledge, the MCVRP addressed in this paper is studied for the first time. The first assumption, is that a fleet of vehicles equipped with multiple compartments, serve customers demands and return to the depot. Each compartment is dedicated to one product. Indeed, a product may regroup different kinds of goods which share the same characteristics. This case is encountered in the real life application which consists in the delivery of groceries to convenience stores. The second assumption, is to serve a customer within a given time windows at minimum cost and a maximum collected profit. This implies that the vehicles can not visit all customers.

In this paper, we propose an hybrid approach based on the Genetic Algorithm (GA) and the Iterated Local Search (ILS), to solve the MCVRP with time windows and profit. This approach exploits the usefulness of both the GA and ILS.

The remainder of this paper is organized as follows. In section 2, the problem formulation is addressed. In section 3, the hybridization approach based on the GA and and ILS is presented. In section 4, preliminary results are presented. Concluding remarks and future works are reported in section 5.

2 PROBLEM FORMULATION

The multi-compartment vehicle routing problem with time windows and profit (MCVRPTW with profit) proposed can be defined as an undirected complete graph G(N,E), in which N={0,...,n} represents the set of nodes and E the set of edges. The depot is represented by the node 0, and a set of customers is represented by the set N'. In this definition, a non negative routing cost c_{ij} and a travel time t_{ij} are associated with each arc (i,j) $\in$ E. A fleet V=$\{1,...,v\}$ of identical vehicles with m compartments, deliver a set P=$\{1,...,m\}$ of m products. Each product p is loaded in compartment p which has a known capacity Q_p. Each customer i has a known request $d_{ip} \leq Q_p$ for each product p. The request of each product to each customer must be delivered by only one vehicle. However, the different products required by one customer can be brought by different vehicles. The customer is served only within a given time windows $[a_i, b_i]$ satisfying $a_i - b_i \geq s_i$, where s_i denotes the service time. A product is delivered to a customer if and only if a preset lower bound on the collected profit is exceeded.

$$min \sum_{i,j} \sum_{k \in V} c_{ij} x_{ijk} \tag{1}$$

$$s.t \sum_{i \in N} x_{ijk} \leq 1 \quad j \in N', k \in V, \tag{2}$$

$$\sum_{i \in N} x_{ijk} = \sum_{i \in N} x_{jik} \quad j \in N', k \in V, \tag{3}$$

$$\sum_{i,j \in S} x_{ijk} \leq |S| - 1 \quad k \in V, S \subseteq N', |S| \geq 2, \tag{4}$$

$$y_{jkp} \leq \sum_{i \in N} x_{ijk} \quad j \in N', k \in V, p \in P, \tag{5}$$

$$\sum_{k \in V} y_{jkp} = 1 \quad j \in N', p \in P, \tag{6}$$

$$\sum_{j \in N'} y_{jkp}\, d_{jp} \leq Q_p \quad k \in V, p \in P, \tag{7}$$

$$s_{ik} + t_{ij} - K(1 - x_{ijk}) \leq s_{jk} \quad i, j \in N', k \in V, \tag{8}$$

$$a_i \leq s_{ik} \leq b_i, \quad i \in N, k \in V, \tag{9}$$

$$\sum_{k \in V} \pi_{jp} y_{jk} \geq p_{min} \quad j \in N', p \in P \tag{10}$$

$$x_{ijk} \in \{0,1\} \quad i, j \in N, i \neq j, k \in V, \tag{11}$$

$$y_{jkp} \in \{0,1\} \quad j \in N', k \in V, p \in P, d_{jp} \neq 0. \tag{12}$$

The objective function (1) minimizes the total routing cost. Constraints (2) ensure that each customer may be visited at most once by each route. Constraints (3) ensure the continuity of each route: if a vehicle visits node j it must leave it. Constraints (4) define the classical subtour elimination constraints. Constraints (5) set y_{jkp} to zero for each product p if vehicle k do not serve customer j. Due to constraints (6), each product required by a customer is brought by one single vehicle. Constraints (7) ensure the vehicle compartment capacities are respected. Constraints (8) states that a vehicle k cannot arrive at j before s_{ik} + t_{ij} if it is traveling from i to j. Constraints (9) allow customer deliveries within a given time windows. Constraints (10) ensure that a customer is visited if and only if the collected profit exceed a given profit lower bound. Constraints (11) set x_{ijk} to 1 if and only if edge (i,j) is traversed by vehicle k. Constraint (12) set y_{jkp} to 1 if and only if customer j receives product p from vehicle k, 0 otherwise.

3 HYBRID APPROACH

In this section, two meta-heuristics will be combined (e.g. The Genetic Algorithm (GA) and the Iterated Local Search (ILS)), to identify a good feasible solution to our MCVRP with time windows and profit.

3.1 Genetic Algorithm

Genetic Algorithms (GA) are general-purpose search algorithms that use principles inspired by natural population genetics to evolve solutions to problems (Herrera et al., 1998). The GA is used since large number of studies is adopting the GA in routing problem. The GA is in fact a population based meta-heuristic. It is based on the natural mechanism applied to a set of chromosomes (e.g. candidate solution). Indeed, the algorithm starts with the generation of the initial population. In each iteration, the chromosomes are evaluated based on a fitness function and then two chromosomes are selected by using a selection method. After that, the selected chromosomes are altered using genetic operators. The first one is crossover which combines two parents to create two new offsprings. The second one is mutation which alters one or more gene from the offspring to ensure the population diversity. At the end, the new offsprings are evaluated and inserted back in the population. This process is repeated until a stop condition is met.

3.2 Iterated Local Search

The Iterated Local Search (ILS) approach, introduced by Lourenco et al. in 2003, is used to solve several combinatorial optimization problems, especially the routing problems. Compared with other meta-heuristics such as, Particle Swarm Optimization (PSO), Simulated Annealing (SA) and Genetic Algorithm (GA), the ILS approach seems to be the most competitive. The ILS alternates the local search phase around the incumbent solution and the perturbation phase, to diversify the search and to escape from local optima. The algorithm starts its search from an initial solution S_0 obtained randomly or returned by a greedy heuristic. The local search procedure is build to obtain a first local optimum. Iteratively, the perturbation is applied to the current local optimum in order to improve the solution. The algorithm is stopped when a termination condition is met. The pseudo code of the ILS approach (Grosso et al., 2009) is depicted in the following algorithm:

```
Begin
    s0= Generate initial solution;
    s*= Local Search (s0);
    repeat
    s′= Perturbation (s*,history);
    s*′= Local Search (s′);
    s*= Acceptance criterion (s*,s*′,history);
    until termination condition is met;
End.
```

3.3 Genetic Algorithm with Iterated Local Search

We propose an hybrid approach combining the GA and the ILS. Our approach is argued by the fact that the GA may not converge to a global optimum, in addition the Local Search procedure may quickly fall in a local optimum. The main idea behind the hybridization is to improve the GA solutions by using an ILS to intensify the search space. In what follows we will use the GA to find the best solution having the least traveling cost and which satisfy the temporal and capacity constraints, whereas the ILS is used to refine the GA solution and consider the collected profit constraint. The pseudo code of the hybrid approach is as follows:

```
Begin
  Initialize population
  Evaluate each candidate;
          repeat
  Select parents;
          Recombine pairs of parents: Crossover;
          Mutate the resulting offspring;
          Apply ILS to the offspring;
          Evaluate new candidates
          Replacement candidates for the next generation
          until termination condition is met;
End.
```

3.3.1 Solution Representation

When designing a GA, the representation of each candidate (e.g.chromosome) in the population is an elementary point. In our approach a solution x (i.e. chromosome, candidate) is represented by using a vector V(x). This vector is a permutation of nodes (customers), which tries to insert an order in the current route, while non violating the temporal and capacity constraints.

3.3.2 Selection and Crossover

The selection of chromosomes is a crucial step for the

the behaviour of the GA process to generate a new population. Several selection methods have been proposed in the literature, for instance the roulette wheel selection, binary tournament method, stochastic universal sampling, rank selection and some others. To randomly select a parent seems to be a non efficient method. Hence, we use the same selection method introduced by Reeves in 1995 (Reeves, 1995). In this method, a parent is selected according to the following probability distribution:

$$P(S)=\frac{2S}{M(M+1)}$$

where S is the S-th chromosome ranked in a descending order of its objective value (e.g. traveling cost) and M is the population size. In other words, the chromosome with the best objective value (e.g. the least cost) has a higher probability to be selected. After the selection of two parents P1 and P2, the crossover operation tries to swap parts of two parents in the population to generate new offsprings (Derbali et al., 2012). There are many ways to apply the crossover, and which may affect the performance of the GA. In our approach , one cutting point i such that $1 \leq i \leq \rho$ is randomly drawn where ρ is the number of requests. The subsequence of a chromosome is copied from the first parent till this point, the second parent is than scanned and the missed customers are added (Derbali et al., 2012). An example of the crossover operation with a crossover point cp = 3 is depicted by figure.1.

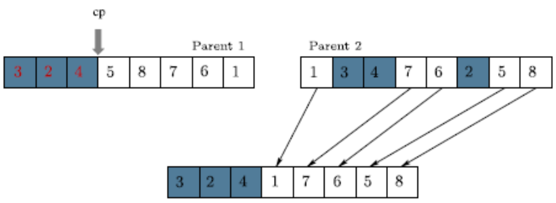

Figure 1: Example of crossover operation.

3.3.3 Mutation

This operator is important to escape from local optima since it contributes to the diversification in the population . The mutation process consists in removing a customer at random and then adding a random customer at a random position. This may lead to explore new solutions .

3.3.4 Local Search

The Local search (LS) is the first step in an Iterated Local Search (ILS) approach. The initial solution of the LS is chosen randomly or by using a metaheuristic. The LS consists in generating a local optimum solution due to the neighborhood exploration of the initial solution. The choice of the neighborhood is one of the important parameter when designing a LS. The neighborhood can be defined as a modification of the initial solution to reach a new better solution. The main focus of the LS in our approach is to find a solution with a minimum traveling cost and a collected profit not low than a preset profit bound. In other words, we will search for solutions that satisfy the profit constraint. In our hybrid approach, we use three neighborhoods including insertion move, swap move and relocate. The neighborhoods are defined as follows:

For the neighborhood *N1*, we insert a request which is not satisfied in the current solution, and which have a collected profit better than the worst profit of a given customer in the current solution. Given a route, the nodes which violate the profit constraint are removed and replace by a new sequence of requests.

Neighborhood *N2* applies the 2-opt move, which consists of replacing two non-adjacent nodes belonging to two different routes, by two non visited nodes. The temporal, capacity and profit constraints should be satisfied.

Neighborhood N3 address the swap move inside the same route. More precisely, we consider one route and swap the position of two nodes belonging to this route.

After the definition of the LS neighborhoods, the LS process is as follows: given an initial solution x (e.g. provided by the GA), we improve x according to *N1* until we found a solution x1 that can not be improved. x1 is the generated local optimum which is considered as the initial solution for the new LS using *N2*. We proceed in the same way with *N3*. This process is repeated until a local optimum is reached.

3.3.5 Perturbation

The first motivation of the ILS algorithm is based on the perturbation operator which is a large random move of the current solution. In fact, the process of the perturbation consists in modifying the current local optimum found by the LS, in order to obtain a new solution. We modify a solution by applying the swap move. We interchange two random selected customers (e.g.nodes) belonging to two different routes.

3.3.6 Evaluation Function

In any GA approach, the chromosomes are compared thanks to their evaluation functions. In this paper, this function is the total traveling cost of a vehicle which leaves the depot, serves customers demands and return to the depot.

3.3.7 Replacement

The replacement operation consists in keeping the population size fixed. It is the last operation in the hybrid approach. After the creation of the offspring by using all the previous phases (i.e GA operators and ILS phases), it is compared with the worst chromosome in the population and then the best one is kept.

4 PRELIMINARY RESULTS

The proposed approach will be tested on a set of instances issued from the well known Solomon's data set (Solomon, 1987). Since no MCVRP with time windows instances are available in the literature, a known instances have been transformed to deal with our proposed approach. In fact, the data set contains 56 problems divided into 6 instances: C1, R1, RC1,C2, R2 and RC2. Each instance contains 100 customers and a depot. Each customer i has an associated demand d_i and a time windows $[a_i, b_i]$. The proposed MCVRP with time windows and profit instances are generated as follows. The number of compartments m for each vehicle is set to 2. The customer request is divided randomly into two parts (El Fallahi et al., 2008). For each customer i the request for product 1 is calculated as $d_{i1} = d_i/\lambda$, where λ is a random integer in $[3,5]$. The request for product 2 is calculated as $d_{i2} = d_i - d_{i1}$. The compartment capacity is computed as $Q_p = (Q \times D_n)/(D_1 + D_2)$, where Q is the vehicle capacity in the original VRPTW (Solomon, 1987) and D_p is the average demand for a product p. The profit π_{ip} is set to 1 for each request. The number of the vehicles is equal to 3.
The proposed approach will be coded in C and will be tested on an Intel Core I5 with 1.70 GHz. The experiments will be carried out and we will compare our proposed approach with some results from the literature, in order to assess the efficiency of the proposed approach.

Table 1 presents the average travel costs results obtained for each instance and the average CPU time for all the instances according to the number of vehicles. The first column contains the name of the instance and the other columns contain the travel cost for one, two and three vehicles. The table 1 shows that the travel cost value increases when the number of vehicles increases and that the problem can be solved within a reasonable time.
It is difficult to provide a detailed analysis at this moment since there are no solutions to compare with. However, in comparison with the solutions found by Fallahi et al. (2008) (e.g. the average cost is equal to 998.645 and the average time is equal to 140.28), our preliminary results are close to the optimal best known solutions. Further extensive experiments will be carried out and we will compare our proposed approach with some results from the literature, in order to assess the efficiency of the proposed approach.

Table 1: Average results of the MCVRP with time windows and profit.

Instances	Cost(m=1)	Cost(m=2)	Cost(m=3)
C1	42.3	612.2	756
C2	1170.7	1228.4	1331.1
R1	924.2	1055.7	1225.7
R2	559.8	759.2	929.3
RC1	752.3	932.1	796.1
RC2	870.8	998.4	1200.2
Avg cost	803.35	931	1093.73
Avg Time(s)	87.7	145.5	220.8

5 CONCLUSION

This paper introduces a research work dealing with the MCVRP with time windows and profit as constraints. Despite its important practical applications, this problem is not yet studied in the literature. It is encountered in a daily distribution of groceries to convenience stores. The proposed work presents a first study, in terms of problem formulation (i.e. to consider time windows and collected profit as constraints), and in terms of problem resolution. In fact, an hybrid approach was developed to address the problem. The mixture exploits the usefulness of ILS in diversifying the genetic search and improving the GA population. More precisely, The GA is used to generate a solution with a minimum total traveling cost, while an ILS is used to optimize the latter solution, in which the temporal, capacity and profit constraints are satisfied. To asses the efficiency, of our proposed approach, we will compare it to a well-known benchmarks from the literature. Further research will be focused on more experiments and comparison with other solution approaches.

REFERENCES

Aras N., Aksen D., Tekin M. T., (2011). Selective multi-depot vehicle routing problem with pricing. Transportation Research Part C: Emerging Technologies.

Chajakis E. D., Guignard M., (2003). Scheduling deliveries in vehicles with multiple compartments. *Journal of Global Optimization*.

Cornillier F., Laporte G., Boctor F. F., Renaud J., (2009). The petrol station replenishment problem with time windows. *Computers & Operations Research.*

Cornillier F., Boctor F. F., Laporte G., Renaud J., (2008). A heuristic for the multi-period petrol station replenishment problem. *European Journal of Operational Research*.

Cornillier F., Boctor F. F., Renaud J., (2012). Heuristics for the multi-depot petrol station replenishment problem with time windows. *European Journal of Operational Research*.

Derbali H., Jarboui B., Hanafi S., Chabchoub H., (2012). Genetic algorithm with iterated local search for solving a location-routing problem. *Expert System with Applications*.

El Fallahi A., Prins Ch, Calvo R. W. (2008). A memetic algorithm and a tabu search for the multi-compartment vehicle routing problem. *Computers & Operations Research*.

Feillet D., Dejax P., Gendreau M. (2001). Traveling Salesman problems with profits: An overview. *Transportation Science*.

Grosso A., Jamali A., Locatelli M., (2009). Finding maximin latin hypercube designs by iterated local search heuristics. *European Journal of Operational Research*.

Herrera F, Lozano M, Verdegay J. L., (1998). Tackling Real-Coded Genetic Algorithms: Operators and Tools for Behavioral Analysis. *Artificial Intelligence Review*.

Hvattum L. M., Fagerholt K., Armentano V. A., (2009). Tank allocation problems in maritime bulkshipping. *Computers & Operations Research*.

Lahyani R., Coelho L. C., Khemakhem M., Laporte G., (2015). A multi-compartment vehicle problem arising in the collection of olive oil in Tunisia. *Omega*

Laurenco M., Martin O., Stutzle T., (2003). Iterated local search. Handbook of metaheuristics of international series in operations research & management science.

Muyldermans L., Pang G. (2010). On the benefits of co-collection: Experiments with a multi-compartment vehicle routing algorithm. *European Journal of Operational Research*.

Oppen J., Lokketangen A. (2008). A tabu search approach for the livestock collection problem. *Computers & Operations Research*.

Reed M., Yiannakou A., Evering R. (2014). An ant colony algorithm for the multi-compartment vehicle routing problem. *Applied Soft Computing*.

Reeves, C. (1995). Modern heuristic techniques for combinatorial problems. *McGraw-Hill Book Company Inc*.

Relvas S., Magatão S. N. B., Barbosa-Póvoa A. P. F. D., Neves Jr. F., (2014). Integrated scheduling and inventory management of an oil products distribution system. *Omega*.

Solomon M. M., (1987). Algorithms for the vehicle routing and scheduling problems with time window and constraints. *Operations Research*.

Stalhane M., Rakke J. G., Moe C. R., Andersson H., Christiansen M., Fagerholt K., (2012). A construction and improvement heuristic for a liquefied natural gas inventory routing problem. *Computers & Industrial Engineering*.

Valle C. A., Martinez L. C., Da Cunha A. S., Mateus G. R. (2011). Heuristic and exact algorithms for a min-max selective vehicle routing problem. *Computers & Operations Research*.