# The Vehicle Routing Problem with Occasional Drivers

**Dr. Shruti Dubey**[1]**, Avijit Maiti**[2]**, Buddhadev Pramanik**[3]**, and Dinesh Samanta**[4]

[1]**Department of Mathematics**
[2]**IIT Madras**

## ABSTRACT

We consider a setting in which a company not only has a fleet of capacitated vehicles and drivers available to make deliveries, but may also use the services of occasional drivers who are willing to make a single delivery using their own vehicle in return for a small compensation if the delivery location is not too far from their own destination. The company seeks to make all the deliveries at minimum total cost, i.e., the cost associated with its own vehicles and drivers plus the compensation paid to the occasional drivers. The option to use occasional drivers to make deliveries gives rise to a new and interesting variant of the classical capacitated vehicle routing problem. We design and implement a solutions by solving an integer programming formulation . A comprehensive computational study provides valuable insight into the potential of using occasional drivers to reduce delivery costs, focusing primarily on the number and flexibility of occasional drivers and the compensation scheme employed.

Keywords: Vehicle routing problem, Occasional drivers

## INTRODUCTION

The VRPOD admittedly does not capture all aspects of the practical problem, but it does allow us to gain quantitative insights in its potential benefits. Specifically, it allows us to study the impact of the number of occasional drivers, the flexibility of the occasional drivers, and the employed compensation scheme on the benefits, i.e., cost savings, for a company. The main limitations of this variant of the vehicle routing problem are that it assumes an occasional driver can only make a single delivery and that it is a static problem. In reality, occasional drivers may be willing (and able) to make multiple deliveries and the availability of occasional drivers will vary over time. However, we believe that, despite these limitations, the insights obtained are valuable and informative.

In order to solve the VRPOD and generate the insight that we seek, we design and implement optimal solutions obtained by solving an integer programming formulation.

Our computational study shows that employing occasional drivers can have significant benefits, and that choosing an appropriate compensation scheme is challenging, in fact more challenging than we initially anticipated. Obviously, the compensation scheme influences the number of available occasional drivers and their willingness to deviate from their intended travel route, but it also impacts which delivery locations will be assigned to occasional drivers (as opposed to the company's drivers), which, in turn, affects the cost savings.

## AN INTEGER PROGRAMMING FORMULATION FOR THE VEHICLE ROUTING PROBLEM

The basic variant of the VRP is defined as follows .

Let G = (N , A) denote a complete directed graph With node set N and arc set A . The node set N is hold two sets of nodes :-

O = The location of the store or depot .

C = The location of the customers.

Here A = $\{(i,j) \in N^2 : i \neq j\}$

Each arc in A has a length and a cost . Each customer i $\in$ C has a given demand $q_i$ and the capacity of vehicle is Q .
Each driver's starting and ending points at depot or store .

Now let $x_{ij}$ be a binary variable indicating whether a vehicle traverses arc (i , j)$\in$A .

$u_i$ be the cumulative demand upto that point i in N .

$c_{ij}$ be the cost of travel over arc (i , j)$\in$A .

The VRP can be formulated as follows

$$Min \sum_{(i,j)\in A} c_{ij}x_{ij}$$

$$\sum_{j\in N, j\neq i} x_{ij} = 1 \;, \qquad \forall i \in C$$

$$\sum_{i\in N, i\neq j} x_{ij} = 1 \;, \qquad \forall j \in C$$

$$\text{If } x_{ij} = 1 \Rightarrow u_i + q_j = u_j \;, \qquad (i,j) \in A \;, i \neq 0 \;, j \neq 0$$

$$q_i \leq u_i \leq Q \;, \qquad \forall i \in C$$

$$x_{ij} \in \{0,1\} \text{ for } (i,j) \in A$$

## THE VEHICLE ROUTING PROBLEM DEFINITION

The basic variant of the VRPOD is defined as follows. Let G = (N, A ) denote a complete directed graph with node set N and arc set A . The node set N is comprised of three sets of nodes: the location of the store or depot (node 0), the locations of the customers (set C ), and the locations of the destinations of the occasional drivers (set K ). For the sake of simplicity, we identify occasional drivers with their destinations. (We assume that an in store customer announces his willingness to deliver goods ordered by an online customer after arriving at the store, and, thus, that the origin of all occasional drivers is the depot.) Each arc a  A has a length $d_a$ and a cost $c_a$ . Each customer i  C has a given demand $q_i$. Regular drivers can make deliveries to customers using a vehicle of capacity Q . We assume that an unlimited number of regular drivers is available. This is consistent with practical applications where companies typically have a sufficient number of drivers to serve all customer requests. An occasional driver k $\in$ K is willing to make a delivery at customer i when $d_{0i} + d_{ik} \leq \zeta d_{0k}$ with $\zeta \geq 1$, i.e., when the extra distance traveled to reach the occasional driver's destination is less than or equal to ($\zeta$-1) times the direct distance from the depot to the occasional driver's destination. An occasional driver can make at most one delivery to a customer. The demand of a customer has to be satisfied either on a route carried out by a regular driver (starting and ending at the depot) or on a trip carried out by an occasional driver, where a trip consists of two parts:-

Traveling from the depot to the customer's location and traveling from the customer's location to the driver's destination. An occasional driver receives $\rho c_{0i}$ as compensation for making a delivery to customer i with $0 < \rho < 1$. Regular driver routes need to satisfy a capacity constraint: the total demand of the customers served on a regular driver's route cannot exceed Q . It is implicitly assumed that an occasional driver k can accommodate the demand $q_i$ for all customers the occasional driver is willing to serve. The cost of a regular driver route r is the sum of the costs of the arcs in the route, i.e., $\sum_{a \in r} C_a$ . The objective (from the company's perspective) is to minimize the total costs, that is the sum of the costs incurred by the regular drivers and the cost incurred for compensating occasional drivers. Note that if there are no occasional drivers, the problem becomes the standard capacitated vehicle routing problem (CVRP). Note too that for a given value of $\zeta$, it is easy to determine the set $C_k$ of customers that an occasional driver k is willing to visit.

In this basic variant of the VRPOD, the compensation paid to an occasional driver is independent of the destination of the occasional driver. This has practical advantages, since the company only needs to know the location of its customers, but may not be ideal from the perspective of an occasional driver, as it does not reflect the extra costs incurred by visiting the customer. An alternative is that occasional driver k receives $\rho(c_{0i} + c_{ik} c_{0k})$ with $\rho \geq 1$ for making a delivery to customer i , i.e., the occasional driver is compensated for the extra mileage incurred. This, of course, is more difficult to implement in practice, because it requires knowledge of the occasional driver's destination. Should it be the location of the occasional driver's home? Should the occasional driver declare her/his destination at the time she/he declares her/his willingness to act as occasional driver? How can this be verified? Even though we are aware of the challenges associated with a compensation scheme that depends not only on the location of the customer, but also on the destination of the occasional driver, we believe it is interesting to see if such a compensation scheme offers any advantages (or disadvantages) in terms of cost savings for the company.

## AN INTEGER PROGRAMMING FORMULATION FOR THE VEHICLE ROUTING PROBLEM WITH OCCASIONAL DRIVERS

Mathematical Model:

- $x_{ij} \rightarrow$ be a binary variable(either 1 or 0) indicating whether a regular vehicle traverses arc ( i , j ) .
- $w_{ik} \rightarrow$ be a binary variable(either 1 or 0) indicating whether an occasional driver k servers an customer i .
- $z_i \rightarrow$ be a binary variable(either 1 or 0) indicating whether a customer is visited by a regular driver .
- $c_{ij} \rightarrow$ distance between ith and jth customers including deport .
- N $\rightarrow$ set of all customers.
- oc $\rightarrow$ set of all occasional drivers.
- A $\rightarrow$ set of all pairs where i $\in$ { 0,1,...n } and j $\in$ { 0,1,2,3...n } and i $\neq$ j.
- $p_{ik} \rightarrow$ compensation given to kth driver for serving ith customer. The basic compensation scheme $p_{ik} = p_i \; \forall k \in$ oc i.e. the compensation for delivering to a customer does not depend on destination of the occasional driver.
- $q_i \rightarrow$ demand of customer i .
- Q $\rightarrow$ total capacity of vehicle of regular driver.
- $u_i \rightarrow$ cumulative demand i.e. for a regular driver total demand upto customer from deport .
- $\beta_{ik} \rightarrow$ it takes value 1 if occasional driver k can serve customer i otherwise 0 . Generally an occasional driver can serve a customer if k be destination of a driver and i be location of customer , then distance from deport(deport location taken as 0) to k via i = d(0,i) + d(i,k) and direct distance = d(0,k). So extra distance driver makes = (d(0,i) + d(i,k)) - d(0,k) ). If cost of extra distance is less than compensation then $\beta_{ik}$ = 1 (i.e. can serve) otherwise 0 .

The integer programming formulation for this problem :

***Minimize :*** $\sum_{(i,j)\in A} C_{ij} x_{ij} + \sum_{i\in N} \sum_{k\in oc} P_{ik} W_{ik}$

- $\sum_{j\in V \cdot j\neq i} x_{ij} = z_i \qquad \forall i \in \mathrm{N}$
- $\sum_{i\in V \cdot i\neq j} x_{ij} = z_j \qquad \forall j \in \mathrm{N}$
- if $x_{ij}$ = 1 then $u_i + q_j = u_j$ for (i,j) $\in$ A : i $\neq$ 0 and j $\neq$ 0 .
- $q_i \leq u_i \leq Q for$ i $\in$ N
- $\sum_{k\in oc} w_{ik} + z_i = 1 \; \forall$ i $\in$ N
- $w_{ik} \leq \beta_{ik} \; \forall i \in N \; , \forall k \in oc$
- $\sum_{i\in N} w_{ik} \leq 1 \; \forall$ k $\in$ oc
- $x_{ij}$, $w_{ik}$ ,$z_i$ are binary variables either 0 or 1 .

The objective function aims at minimizing the total cost with respect to constraints. Here first two constraints are flow conservation constraints. Third and fourth are ensure that the vehicle capacity is respected. Fifth ensure that a customer visited by either a regular driver or an occasional driver. Sixth ensure if an occasional driver willing to make delivery then only assign to the service . Seventh for the condition an occasional driver makes only one delivery .

# COMPUTATIONAL STUDY

As mentioned in the introduction, the aim of our research is to gain initial quantitative insights in the potential benefits of crowd- shipping for last-mile delivery. More specifically, we aim to under- stand the impact of the number of occasional drivers, the flexibility of the occasional drivers, and the employed compensation scheme on the potential benefits, i.e., cost savings. To be able to do so, we conducted a series of computational experiments using randomly generated instances, because the use of occasional drivers to make deliveries is still in its infancy and at a conceptual stage, and no real-life data exist.
We used IBM ILOG CPLEX solver as exact solver and algorithms were coded in Anaconda Jupyter Notebook (Python 3) .

## 0.1 Potential benefits of employing occasional drivers

The potential benefits of employing occasional drivers to make deliveries depend on three factors:

- how many occasional drivers there are relative to the number of customers that need to be served .
- how much flexibility an occasional driver has .
- how much an occasional driver is paid for making a delivery, i.e., the compensation scheme and the value of $\rho$ .

To obtain quantitative insights into the impact of these three factors, we solve each instance in the test set of the following parameter values driver customer ratio and compensation $\rho$ = 1, 1.5 .We assess the potential benefits of employing occasional drivers by examining and comparing the total cost, the fraction of the total cost contributed by regular drivers, and the number of occasional drivers employed for the different solutions.In our analysis and comparisons, we also include the solution obtained when no occasional drivers are available (which provides an upper bound on total cost) and the solution obtained .

## 0.2 Tables and Figures

We tested two different compensation scheme for $\rho = 1$ and $\rho = 1.5$ .Since compensation depends on distance so $p_{ik}$ $p_i = \rho$ * d(0,i) (i.e. distance between depot and customer i). Value of $\rho$ decides whether occasional driver is profitable or not .If we take $\rho = 1$ then $p_i$ = d(0,i) and if we take $\rho = 1.5$ then $p_i$ = 1.5 * d(0,i) .

### *0.2.1 Compensation Scheme I*

| No of customers | No of occasional drivers | Total cost | No of occasional drivers used |
|---|---|---:|---:|
| N = 10 | K = 0 | 726.24 | 0 |
| | K = 2 | 671.716 | 2 |
| | K = 5 | 644.876 | 3 |

**Table 1.** scheme I for 10 customers.

| No of customers | No of occasional drivers | Total cost | No of occasional drivers used |
|---|---|---:|---:|
| N = 15 | K = 0 | 741.083 | 0 |
| | K = 3 | 694.534 | 3 |
| | K = 6 | 685.816 | 4 |

**Table 2.** scheme I for 15 customers.

| No of customers | No of occasional drivers | Total cost | No of occasional drivers used |
|---|---|---:|---:|
| N = 20 | K = 0 | 1357.09 | 0 |
| | K = 5 | 1330.46 | 4 |

**Table 3.** scheme I for 20 customers.

### 0.2.2 Compensation Scheme II

| No of customers | No of occasional drivers | Total cost | No of occasional drivers used |
|---|---|---|---|
| N = 10 | K = 0 | 726.24 | 0 |
| | K = 2 | 716.971 | 1 |
| | K = 5 | 716.971 | 1 |

**Table 4.** scheme II for 10 customers.

| No of customers | No of occasional drivers | Total cost | No of occasional drivers used |
|---|---|---|---|
| N = 15 | K = 0 | 741.083 | 0 |
| | K = 3 | 728.811 | 2 |
| | K = 6 | 728.811 | 2 |

**Table 5.** scheme II for 15 customers.

| No of customers | No of occasional drivers | Total cost | No of occasional drivers used |
|---|---|---|---|
| N = 20 | K = 0 | 1373.5 | 0 |
| | K = 5 | 1360.75 | 1 |

**Table 6.** scheme II for 20 customers.

### 0.2.3 Figures

We experiment with various data some of which given above in the tables. We also observed various figures in our computational studies. Here first set of images experimented for 10 customers without occasional drivers and with 5 occasional drivers and compensation scheme I ($\rho = 1$). Second set of images computed for 15 customers without occasional drivers and 5 occasional drivers and compensation scheme II ($\rho = 1.5$) .
In this experiments we observed occasional drivers are used in those areas where customer density(no of customer in this area) is low. In this cases using occasional drivers would be profitable. Also we observed most of cases if compensation scheme is cheap then occasional drivers served in long distanced area and if compensation is not cheap then serves nearby areas but not all the cases its follows.
Also from tables above we can observed that not always increasing of occasional drivers changes the solution, this is because for some customers location occasional driver would be effective but not for all ,that is why increasing drivers does not mean always change solutions.It may be happens that in some cases the model do not consider any occasional drivers, in that case all customers served by regular drivers would be cost effective.

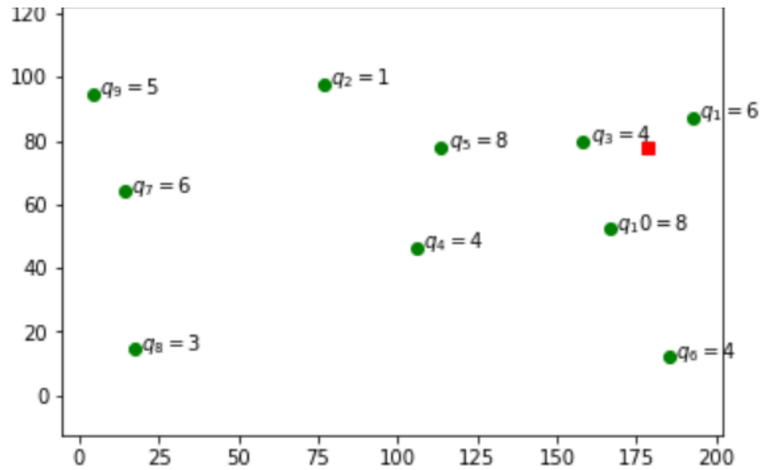


**Figure 1.** Location of customers and deport.

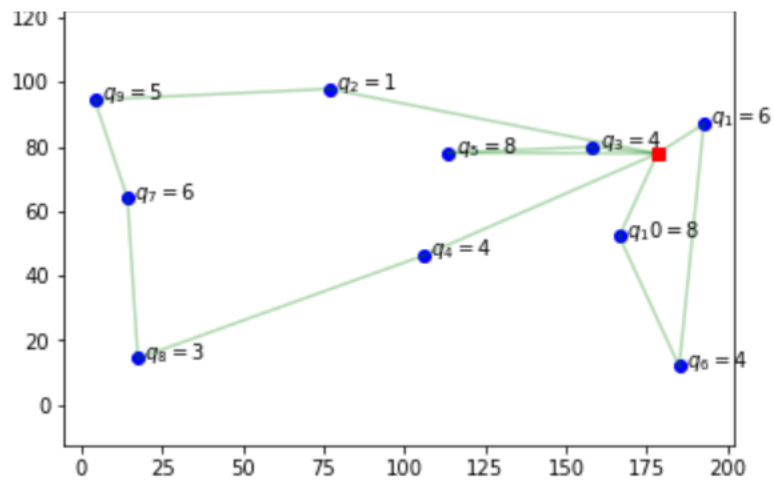


**Figure 2.** VRP without occasional drivers.

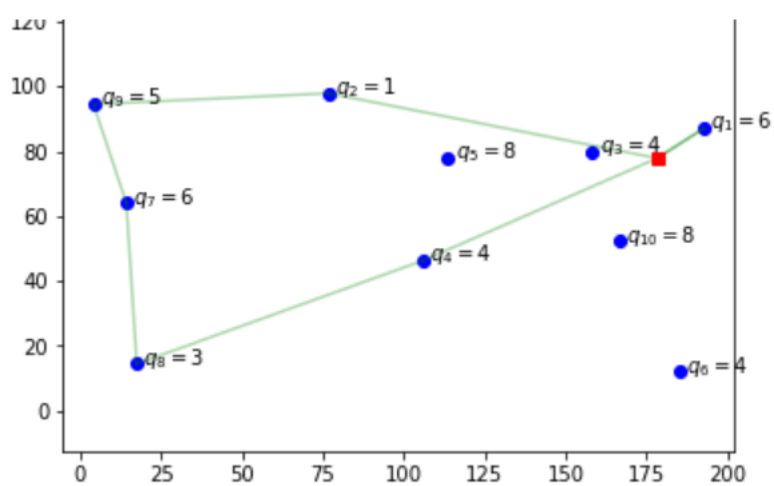


**Figure 3.** VRP with occasional drivers.

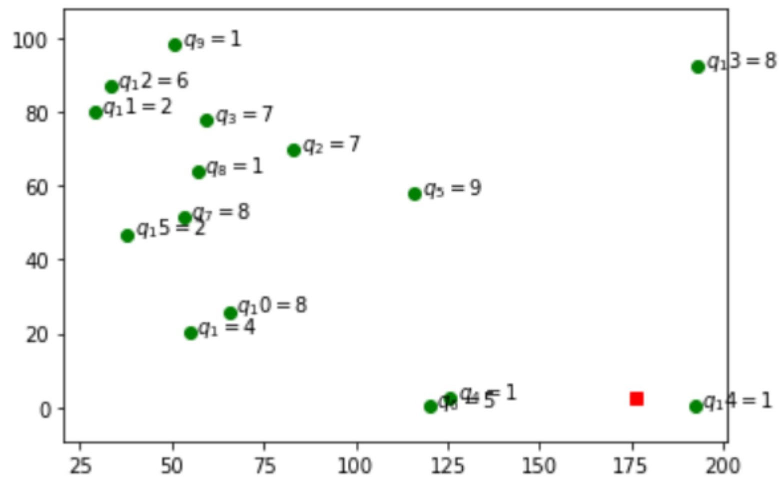


**Figure 4.** Location of customers and deport.

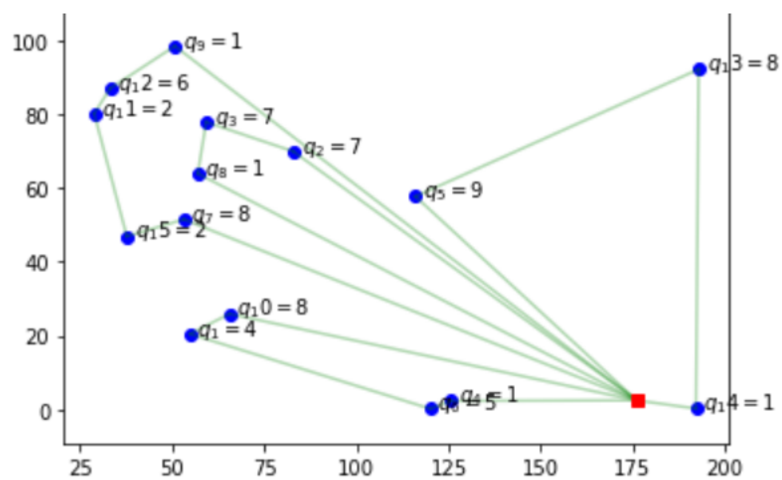


**Figure 5.** VRP without occasional drivers.

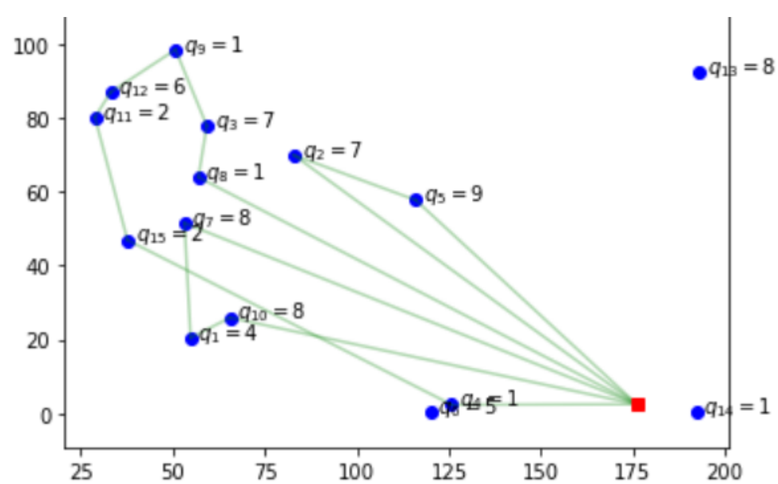


**Figure 6.** VRP with occasional drivers.

## FINAL REMARKS

The goal of our investigation has been to gain an initial understanding of the potential benefits and the implementation challenges associated with crowd shipping. The results of our computational study are both encouraging and dispiriting. We have found that substantial cost savings can be realized when there is a large number of people with a generous amount of flexibility available to make deliveries. That, of course, depends to a large extent on the compensation offered. Designing an appropriate and cost-effective compensation scheme is one of a major implementation challenges associated with crowd shipping. We have experimented with two natural compensation schemes and found that the performance of both of them was acceptable, but sensitive to the choice of parameters. Compensation schemes based on the "cost-to-serve"of a customer may be most appropriate. Of course determining the cost-to-serve of a customer is notoriously difficult in routing problems, and will be even more complicated than usual in this setting as it needs to account for the fact that some of the customers may be served by crowd shippers. Research into more sophisticated compensation schemes is both interesting and necessary. We have assumed that an occasional driver only declares his willingness to make a delivery after arriving at the store (depot). This is reasonable in the context of Walmart, but not in the context of Amazon, who does not operate stores, but only distribution centers. An interesting extension is to study a setting in which occasional drivers offer to make a delivery before leaving the origin of their trip. (This would also be beneficial in the Walmart setting.) Another avenue for further research, maybe even more important and more interesting, is the study of variants of the VRPOD, in which aspects of the highly dynamic nature of the setting are captured. In reality, occasional drivers become available over time and their services will only be available for a short period of time after they become available. Of course, orders from customers become available over time too, and, these orders imply new deliveries . Almost all the literature on dynamic vehicle routing focuses on orders to be picked up, orders to be picked up and delivered, or to a service performed by the driver. The underlying structure of the dynamic routing problem changes significantly when new orders have to be delivered, since there are few, if any, opportunities to accommodate additional deliveries after a delivery vehicle has left the depot, because the vehicle would have to return to the depot to pick up the additional deliveries.

## REFERENCES

[1] Claudia Archetti , Martin Savelsbergh , M.Grazia Speranza . The Vehicle Routing Problem with Occasional Drivers : European Journal of Operational Research 254 (2016) 472–480 .

[2] Claudia Archetti , Francesca Guerriero , Giusy Macrina (2021). The online vehicle routing problem with occasional drivers : ELSEVIER Computers and Operations Research

## A MULTI-START HEURISTIC FOR THE VRPOD

Since integer programming very time consume process and for large set of customers we cannot use this.For this reason here a multi-start heuristic for the solution of the VRPOD ,[1] which combines variable neighbourhood search and tabu search(Integers programs are also solved to optimality to assign customers to occasional drivers, thus the solution approach may be seen as belonging to the class of matheuristics.A high-level overview of MATHHOD, the name we have given to the approach, can be found in algorithm below.

```
Algorithm : MATHOD
I ← constructInitialsolutions;
for all the i ∈ I do
    s ← i;
    k ← 1;
    while k ≠ k_max do
        s ← InternalTabuSearch;
        s ← Jump(k,s);
        if s is better than s_best then
            s_best ← TwoOpt(s);
            s ← s_best;
            k ← 1;
        else
            k ← k+1;
        end
    end
end
```

Next, we describe the procedure that comprise MATHOD in more detail.
ConstructInitialSolutions. MATHOD embeds a multi-start approach with five different initial solutions. Any initial solution is characterized by a set $S \subseteq C$ of customers to be served by regular drivers. The routes for the regular drivers are constructed using a sequential greedy insertion algorithm, which inserts customers into the active route in non-decreasing order of their distance to the depot, starting a new route when the vehicle capacity constraint would be violated, and, after all customers in S have been inserted, improves each route using a 2-exchange neighbourhood.The five different sets S are generated as follows :

1. Simply take S = C, i.e., all customers are served by regular drivers.

2. Solve the following integer program to determine the set of customers to be served by occasional drivers, where as before, $w_{ik}$ is a binary variable indicating whether customer i is visited by occasional driver k and $\beta_{ik}$ indicates whether occasional driver k can serve customer i :

$$max \sum_{k\in K}\sum_{i\in C}(2C_{0i} - p_{ik})w_{ik}$$

$$w_{ik} \leq \beta_{ik} \quad \forall i \in C \quad \forall k \in K \tag{9}$$
$$\textstyle\sum_{i\in C} w_{ik} \leq 1 \quad \forall k \in K \tag{10}$$
$$\textstyle\sum_{i\in K} w_{ik} \leq 1 \quad \forall c \in C \tag{11}$$
$$w_{ik} \in \{0,1\} \quad i \in C \quad k \in K. \tag{12}$$

That is, we seek to serve as many far-away customers with occasional drivers served by an occasional drivers. Let $S^{'}$ be the set of customers served by an occasional driver i.e., for which $\sum_{k\in K} w_{ik} = 1$ in the optimal solution. Then S = C $\backslash S^{'}$.

3.$Solvetheintegerprogram$
$max \sum_{i \in C} \sum_{k \in K} q_i w_{ik}$
subject to constraint (9)-(12). That is, maximize the demand delivered by the occasional drivers. Again, if $S^{'}$ is the set of customers served by an occasional driver in the optimal solution, then S = C $\backslash S^{'}$.

4. Solve the integer program
max $\sum_{i \in C} \sum_{k \in K} q_i w_{ik}$
subject to constraint (9)-(12).Let $\theta$ be the value of an optimal solution. Note that $\theta$ corresponds to the maximum number of occasional drivers that can be engaged to serve customers. next, solve the following integer program :
min $\sum_{i \in C} \sum_{k \in K} p_{ik} w_{ik}$
subject to constraint (9)-(12) and $\sum_{k \in K} w_{ik} = \theta$.
find the minimum cost solution that uses the maximum possible number of occasional driver in the optimal solution, then S = C$\backslash S^{'}$.

5. Solve a relaxation of the integer program presented in the section integer programming formulation obtained by ignoring the integrality constraint on variable $x$. The set S contains those customers i for which $z_i = 1$.

InternalTabuSearch (s). for a given solution s, let $C^{'}$ denote the set of customers served by occasional drivers and let $r_i$,for i $\in$ C$\backslash C^{'}$, denote the route of the regular driver serving customer i. For a customer i and a route r $\neq r_i$, we denote by r+i the route obtained by inserting i into r(using the cheapest insertion criterion to determine where to insert i). Similarly, given a route $r_i$ of a regular driver and a customer i served on $r_i$, we denote $r_i - i$ the route obtained from $r_i$ by deleting i and joining its predecessor with its successor. The tabu search performed on s uses four feasible moves :

I-move : A customer i $\in C \backslash C^{'}$ is moved from its route $r_i$ to a route $r \neq r_i$, where r may be the empty route. Hence, $r_i$ and r are replaced by $r_i - i$ and r+i, respectively.

II-move(swap move) : Customers i,j $\in C \backslash C^{'}$, with $r_i \neq r_j$, are exchanged. Hence, $r_i$ and $r_j$ are replaced by $(r_i - i)$+j and $(r_i - j)$+i, respectively.

III-move(in-move) : A customer i $\in C^{'}$ is inserted into a route r. Hence, r is replaced by r+i.

IV-move(out-move) : A customer i $\in C \backslash C^{'}$ is assigned to an occasional driver (if possible, i.e., their exists an occasional driver k, which does not serve a customer in s and for which $\beta_{ik} = 1$). Hence, $r_i$ is replaced by $r_i - i$.

All non-tabu moves are evaluated and the best one is chosen.Ties are broken arbitrarily.
A temporary tabu status forbids customers to be inserted in routes from which they have been recently removed. Also, when a customer i $\in C^{'}$ (C$\backslash C^{'}$) is moved to C$\backslash C^{'}$($C^{'}$), then it is temporarily tabu to move it back to $\in C^{'}$ (C$\backslash C^{'}$). Each time a new best solution is found, each route is improved with a local search procedure using a 2-exchange neighbourhood. The tabu search terminates after $n_{max}$ iterations without improvement. At each iteration all moves are evaluated and the best non-tabu move is chosen. A tabu move is chosen only if it improves the best solution found so far.

Jump(k,s). For a given solution s, a customer i $\in C^{'}$ is randomly selected and inserted in a route $r$ (using the cheapest insertion criterion to determine where to insert i), if no route can accommodate customer i, then a route is created. The procedure is repeated $min\{k, |C^{'}|\}$ times. Let $C^{''}$ be the set of customers which are still served by an occasional driver plus those customers served by a regular driver in a route which visits a single customer, but excluding those customers that have just been removed from $C^{'}$. Furthermore, let $K^{'}$ be the set of occasional drivers who are not used in the current solution. Then, we solve the following integer program :

max $\sum_{k \in K'} \sum_{i \in C''} \gamma_i w_{ik}$
$w_{ik} \leq \beta_{ik} \quad \forall i \in C'' \quad \forall k \in K'$
$\sum_{i \in C''} w_{ik} \leq 1 \quad \forall k \in K'$
$\sum_{k \in K'} w_{ik} \leq 1 \quad \forall i \in C''$
$w_{ik} \in \{0,1\} \quad i \in C'' \quad k \in K'$,

where $\gamma_i$ is randomly drawn from a uniform distribution over [0,10]. That is, we seek to generate a new assignment of customers to occasional drivers and thus a solution that considerably differs from the one obtained at the end of the previous internal tabu search phase.
TwoOpt (s). This procedure aims at improving each regular driver route using a 2-exchange neighbourhood.

End