Author's preprint version of
Hasan Murat Afsar, Sezin Afsar, Juan José Palacios. *Vehicle routing problem with zone-based pricing.*
Transportation Research Part E: Logistics and Transportation Review,
Volume 152, p.102383. 2021
https://doi.org/10.1016/j.tre.2021.102383.

Vehicle Routing Problem with Zone-Based Pricing

Abstract

In this paper we study the vehicle routing problem with zone-based pricing where the transporter defines a service price for all customers in a geographical zone. Each customer accepts or rejects the transporter's offer according to a personal threshold value. Once the customer accepts the offer, the delivery service becomes compulsory. Two decisions need to be made: the price per zone and the routing to serve all customers that accept that price. The objective is to maximize the profit, which is the difference of total service revenue and the routing costs. We present a mathematical model for the problem and propose a branch-and-price algorithm. A numerical analysis shows promising results on the proposed set of instances.

Keywords: Vehicle Routing Problem with Pricing, Branch-and-Price, Zone-pricing

1. Introduction

The well-known vehicle routing problem (VRP) aims at designing distribution routes so a set of customers can be visited by a fleet of vehicles, typically minimizing the costs or maximizing the profit of the delivery company (Dantzig and Ramser (1959)). The problem has been very relevant in the literature due to its many applications in the real world, not only for delivering goods but also offering services, or collecting different items to redistribute them.

In today's society, companies understand the necessity of complementing provision of industrial goods with value-added services (Legnani et al. (2013)). Customers do not settle for the delivery of goods alone and instead, demand more services and quality of service. Therefore, the customers do not only decide if they want to pay for home delivery, but also they tend to have in mind a price they are willing to pay depending on the services the company offers, as for instance being able to fix a delivery date. One of the main factors involving the delivery fee is the geographical zone of the delivery address, being more expensive to deliver to further areas while keeping a low price for close-by zones. This aligns in general with customers' price expectation, as someone living close to the facility will not be satisfied paying a high price for delivery, whereas a customer living far might find it acceptable. Home delivery has the potential of integrating different socio-economic layers in an urban context. Providing service and being present in all zones contributes to the social well-being. Therefore, the delivery service provider should set the service prices in such a way that at least one customer per zone can afford the service. Hence the delivery service provider has to make two decisions: establishing a service price per geographical zone to increase revenue, and designing vehicles' routing to decrease operational costs.

Notice that these two decisions have an impact on each other. If the service price is too low, many customers will opt for the delivery and this will increase operational costs. On the other hand, if the service price is too high, much fewer deliveries would be planned, which keeps operational costs low but also reduces the overall profit. Therefore, the objective is to find an optimal trade-off in the sense that the overall profit is maximized.

Different variants of VRP have been tackled in the literature. For instance, different delivery policies have been considered. A delivery service policy is said to be mandatory when serving all customers is obligatory. Under this policy, the total revenue of the provider is fixed and therefore total operational cost is desired to be minimized. On the other hand, the delivery service is said to be selective when serving all customers is optional. In this case, it is possible to focus on serving only a subset of customers to maximize the total profit. Fischetti et al. (2007) underline that in several applications, only a subset of nodes (customers) need to be visited and define Simple Cycle Problem (SCP). They provide a list of such variant problems of SCP and exact solution methods based on branch-and-cut. According to Archetti et al. (2009), many articles in the literature of routing problems focus on mandatory service whereas the extent of papers that study profits is much more limited.

Regarding routing problems with profits, variants are divided into three categories by Feillet et al. (2005) depending on how the profits and travel distances/costs are managed. In *prize collecting* routing problems, a lower bound is set on the amount of prize to be collected while minimizing the total distance traveled (Balas, 1989). The *orienteering problem* defined by Golden et al. (1987); Tsiligirides (1984), targets at maximizing the total collected profit while respecting a fixed budget on the travel costs. This might be considered the most popular variant and it is also referred to as selective traveling salesman problem (Laporte and Martello, 1990). For instance, the survey of Vansteenwegen et al. (2011) focuses on orienteering problems inspired by a competition where each node has a score, and the competitors should obtain a maximum total score within a time horizon. In their work, classical forms and extensions such as *team orienteering* and *orienteering with time windows* are tackled with popular solution methods. More recently, several variants of routing problems with profits, their mathematical models and solution approaches are presented in a tutorial in (Vansteenwegen and Gunawan, 2019). The authors also mention several interesting applications such as athlete recruitment, mobile crowd sourcing problem and wildfire routing problem. In many variants, the profit to be collected at each customer is fixed beforehand and it has to be collected all at once. In some other cases, the collected profit depends on the time spent on the node (Erdogan and Laporte, 2013), travel time (Afsar and Labadie, 2013) or the position of the visited node in the route (Reyes-Rubiano et al., 2020).

Lastly, the *profitable tour* problem (PTP) optimizes the combination of collected total profit and total traveling cost (Dell'Amico et al., 1995). Jepsen et al. (2014) recognize the capacitated version of PTP as the sub-problem in Dantzig-Wolfe decomposition of many routing problems and propose a branch-and-cut framework.

Liu and Chen (2011) observe the impact of the pricing decisions on demand and study inventory routing and pricing in a supply chain. The authors consider a linear relation between price and demand and propose a heuristic strategy to maximize the total revenue in this scenario. This approach randomly chooses at each iteration to improve either inventory routing solution or pricing. Etebari and Dabiri (2016) extend this work by setting the prices dynamically according to the period and designing a simulated annealing framework with five phases embedded. In this work, the relation between price and demand is also considered as linear. Furthermore, it is assumed that the willingness to pay for a service may be time dependent. It is argued that for a customer, the value of seasonal goods such as perishable food or liquefied gas, can change from season to season.

In a reverse supply chain context, Aras et al. (2011) work on a selective multi-depot vehicle routing problem in which a firm buying goods from dealers sets a price. The purchase is done if this price is above the dealer's threshold and the firm decides that the deal is profitable. The authors provide two linear model formulations and a tabu search heuristic to solve this problem.

Ahmadi-Javid and Ghandali (2014) study price-sensitive demands in a distribution network under both mandatory and selective service policies of all clients. In this work, a mixed integer linear programming model is solved by a Lagrangian Relaxation procedure. The location-allocation model is then expanded by location inventory and pricing decisions in (Ahmadi-Javid and Hoseinpour, 2015).

More recently, a branch-and-price algorithm was used by Ahmadi-Javid et al. (2018) to solve a location routing problem with pricing decisions where discrete price and corresponding demand levels are considered under a selective service policy. At first glance, this problem seems to be similar to ours, but there are structural differences that differentiate the models, even though a branch-and-price scheme is employed in both cases. Setting a price per customer, a demand depending on the price level and non-mandatory service conditions create serious dissimilarities in the solution of the sub-problem and branching schemes.

The main contributions of this work are cited below.

- We present a novel, and more realistic, version of the Vehicle Routing Problem with price setting where the transporter sets a price for each geographical zone rather than each customer. We consider zone-pricing a realistic approach since most home-delivery service providers in city logistics context tend to divide the city into concentric zones where the depot is in the middle. It is also natural that each customer has a threshold or a maximum price they are willing to pay for the service. If the price is lower than their threshold, they accept the delivery, and they opt for picking up the package themselves if it is higher. Note that if the customer accepts the price, the deal makes it compulsory for the transporter to deliver the goods. It is important to emphasize that for every set of zone-pricing decisions, there is a different set of customers to be served and hence a different VRP to solve. Since VRP is *NP-hard*, so is our problem.

- We demonstrate that, although the price for each zone can take any value, considering only customer threshold values is dominant. This property is crucial for establishing the equivalence between the original formulation and the one with a linearized objective function.

- We solve this problem optimally by a branch-and-price algorithm. Thanks to the previous property, visit variables are replaced with threshold setting variables. This leads us to separate price and cost calculations and simplify the sub-problem.

- An adequate branching scheme is applied which takes into account fleet size, price setting and flows on arcs.

- As a result of the price setting, we propose a graph reduction which accelerates the solution of the sub-problem.

The paper is organized as follows: the notation to be used and a formal definition of the problem are introduced in section 2. The set packing formulation, the sub-problem and the branching procedure for a branch-and-price algorithm are given in section 3. In section 4, a numerical analysis is presented. Concluding remarks and some future research directions are provided in section 5.

2. Problem Statement and Formulation

In this section we introduce a formal definition of the problem and a mathematical model along with the notation.

2.1. Problem Definition

The vehicle routing problem with zone-based pricing (**VRP-ZP**) can be defined on a directed complete graph $G(V, A)$ where V represents the depot node (0) and the potential customers (V^+) considering the home delivery service, and A the arcs connecting every pair of nodes. The set V^+ is divided in p subsets representing distinct zones ($V^+ = V_1 \cup V_2 \cup ... \cup V_p$ such that $V_k \cap V_{k'} = \emptyset \;\; \forall k \neq k'$ and $k, k' \in \{1, ..., p\}^2$). Each customer $i \in V^+$ has a demand q_i to be satisfied and a threshold value th_i.

A fleet of identical vehicles with a capacity of Q is located at the depot (0) in the beginning of a working day. At the end of the working day, all vehicles must return to the depot. Every time a vehicle travels through an arc $a_{i,j} \in A$, a cost $c_{i,j}$ is incurred. In addition, the total amount of goods delivered by each vehicle cannot exceed its capacity.

The transporter can define only one price $p_k \in \mathbb{R}$ for each zone k, but to ensure that no geographical zone is left without service, at least one customer per zone has to be served so that the zone price is set. Each customer i ($i \in V_k$) can be visited at most once and p_k can be collected.

We define a function $z(i)$ that gives the geographical zone of customer i.

$$z(i) = k \iff i \in V_k,\ \forall i \in V^+$$

For the sake of simplicity, the customers in each zone are sorted in an increasing order of their threshold values ($i < j \iff th_i < th_j$ and $z(i) = z(j)$). We define another function $l(i)$ which returns the number of customers in the same zone having a threshold greater than or equal to th_i.

$$l(i) = |\{j\ :\ th_j \geq th_i \text{ and } z(i) = z(j)\}|$$

If the price p_k proposed by the transporter for the zone that the customer i is in exceeds threshold value of this customer, the customer refuses the home delivery.

The objective is to maximize the total profit which is the difference between the total price paid by the customers who accept to be served and the travel costs induced by the delivery of their goods.

In the following we show that, even if the transporter is free to choose any value for the price p_k for a zone k, in the optimal solution, that value should be equal to the threshold value of one of the customers of this zone.

Proposition 1. *In the optimal solution, $\forall k, p_k \in \{th_i \,:\, i \in V_k\}$.*

Proof. Let us assume that, there exists a geographical zone k where the optimal price $p^*_k = \bar{p_k}$ is such that $th_i < \bar{p_k} < th_{i+1}$. As the transporter should serve all the customers having a greater threshold value than the price proposed, and the customers of each zone are sorted in an increasing order of their threshold values, the total price to be collected from this zone is $\bar{P_k} = \bar{p_k} \times l(i+1)$. However, if we take $p_k = th_{i+1}$, total price of this zone would be $\bar{\bar{P_k}} = th_{i+1} \times l(i+1)$. It is trivial that $\bar{\bar{P_k}} > \bar{P_k}$ and the set of customers to be served remains the same, therefore the travel costs are the same as well. So, in the second case, total profit obtained is strictly greater than the first one, which contradicts with the initial statement. □

2.2. Mathematical Model

We define a binary variable $x_{i,j}$ taking value 1 if the arc (i, j) is traversed by a vehicle. A real-valued variable f_i corresponding to the total load delivered by the vehicle leaving the node i. The binary variable w_i is set to 1 if the customer i accepts the service price proposed by the service provider. In that case, the customer should be visited by a vehicle. Let us remember that p_k is the price proposed to the geographical zone k.

$$\max \Theta = \sum_{i \in V^+} w_i \cdot p_{z(i)} - \sum_{i \in V} \sum_{j \in V} c_{i,j} \cdot x_{i,j} \tag{1}$$

Subject to

$$w_i \cdot p_{z(i)} \leq th_i \qquad \forall i \in V^+ \tag{2}$$

$$\sum_{i \in V_k} w_i \geq 1 \qquad \forall k \in \{1, ..., p\} \tag{3}$$

$$\sum_{i \in V} x_{i,j} = w_j \qquad \forall j \in V^+ \tag{4}$$

$$\sum_{i \in V} x_{i,j} - \sum_{i \in V} x_{j,i} = 0 \qquad \forall j \in V \tag{5}$$

$$f_i - f_j + q_j + \bar{Q} \cdot x_{i,j} \leq \bar{Q} \qquad \forall i \in V,\ \forall j \in V^+ \tag{6}$$

$$f_i \leq w_i \cdot Q \qquad \forall i \in V^+ \tag{7}$$

$$f_0 = 0 \tag{8}$$

$$p_k \in \mathbb{R}^+ \qquad \forall k \in \{1, ..., p\} \tag{9}$$

$$f_i \geq 0 \qquad \forall i \in V \tag{10}$$

$$w_i \in \{0, 1\} \qquad \forall i \in V^+ \tag{11}$$

$$x_{i,j} \in \{0, 1\} \qquad \forall (i, j) \in A \tag{12}$$

The objective function (1) maximizes the difference between the total collected price and travel costs. According to the constraints (2), the customers accept to be served if the price proposed for their zone is less than or equal to their threshold value. Constraints (3) force the transporter not to abandon entirely a geographical zone and to serve at least one customer. If a customer agrees to be served, then a vehicle should visit him, following constraints (4). Constraints (5) conserve the flow. Constraints (6) eliminate the subtours. If customer j is served right after customer i (by the same vehicle) the flow should change accordingly. Otherwise, $f_j \geq f_i + q_j - \bar{Q}$ where $\bar{Q}$ is $Q + \max\limits_{v \in V^+} q_v$. As f_i is bounded by Q, and q_j by $\max_{v \in V^+} q_v$, this constraint states that f_i is greater than a negative value, which is always true. The total quantity delivered on a vehicle's route is limited by its capacity in constraints (7). Note that constraints (6) always hold for any pair of customers i, j even when they are not served consecutively by the same vehicle (i.e. $x_{ij} = 0$) due to constraints (7). Finally, constraint (8) ensures that the total distributed quantity is zero at the beginning of any route, and constraints (9 - 12) define the variables' domains.

Notice that that this mathematical model is not linear since two variables are multiplied in the objective function and also in constraints (2). However, proposition 1 can be exploited to linearize the model. By defining a binary variable y_i that takes value 1 *iff* $p_{z(i)} = th_i$, the objective function can be reformulated as:

$$\max \Theta = \sum_{i \in V^+} y_i \cdot th_i \cdot l(i) - \sum_{i \in V} \sum_{j \in V} c_{i,j} \cdot x_{i,j} \tag{13}$$

If the price of a zone k is set to the threshold value of a customer i, then the transporter can collect th_i as the price from each of the $l(i)$ customers of this zone. To ensure that one price is set per zone, an additional constraint has to be added:

$$\sum_{i \in V_k} y_i = 1 \ \forall k \in \{1, ..., p\} \tag{14}$$

This constraint also ensures that at least one customer per zone is served, thus constraint (3) is now redundant and can be removed. Finally, constraints

(2) can be redefined by replacing w_i as follows:

$$w_i = \sum_{j \in V_{z(i)} \,:\, th_j \leq th_i} y_j \quad \forall i \in V^+ \tag{15}$$

This constraint verifies that if the price for the zone k of customer i ($k = z(i)$) is set to the threshold th_j of a customer such that $th_j < th_i$, then customer i has to be visited.

Since the objective function (1) and the constraints (2) are replaced, the variable p_k is dropped as well. Despite the resulting model being linear, it is still too complex to solve, and even small instances cannot be solved by a commercial solver in a reasonable amount of time, which is explained further in section 4. Therefore, we present a Set Packing formulation for this problem in the following section.

3. Branch-and-Price Algorithm

To be able to solve the VRP-ZP, we propose a Dantzig-Wolfe decomposition with an exponential number of variables (Dantzig and Wolfe (1960)). There are several applications in the literature of Dantzig-Wolfe decomposition and its solution by Column Generation. Among others, it is applied in Boussier et al. (2007) for team orienteering problems, in Desaulniers et al. (2016) for electrical vehicle routing problems, in Ponboon et al. (2016) for location routing problems or in Liu et al. (2018) for multiple repairmen problems.

More recently, Faiz et al. (2019) employ Column Generation for a variant of open VRP with time windows. With an improved branch-and-price algorithm, Yu et al. (2019) solve a Green Vehicle Routing Problem with time windows and heterogeneous fleet. In the sub-problem, they evaluate the labels for all types of vehicles and reduce considerably number of labels. Tas (2021) applies the Column Generation to the Electric Vehicle Routing Problem with soft time windows. If the final solution is not integer, the set partitioning model is solved with the integrity constraints.

3.1. Master Problem

The Dantzig-Wolfe decomposition of the VRP-ZP allows us to reformulate the previous model by a route-based presentation. Let λ_r be a binary variable taking value 1 if the feasible route $r \in \Omega$ is in the optimal solution, where Ω is the set of all feasible routes. Notice that the size of Ω would be too large to be able to solve any practical instance. To overcome this, we shall start with a restricted set Ω' such that $|\Omega'| << |\Omega|$ and dynamically add feasible routes, and their corresponding variables, with a column generation schema. Let C_r denote the total travel cost of a route r, and let γ_r^i count the number of visits paid to customer i by the route r. A linear relaxation of the restricted master problem (LRMP in short) is formulated as follows:

$$\max \Theta = \sum_{i \in V^+} y_i \cdot th_i \cdot l(i) - \sum_{r \in \Omega'} C_r \cdot \lambda_r \tag{16}$$

Subject to

$$\sum_{i \in V_k} y_i = 1 \qquad \forall k \in \{1, ..., p\} \tag{17}$$

$$\sum_{r \in \Omega'} \gamma_r^i \cdot \lambda_r - \sum_{j \in V_{z(i)} \,:\, th_j \leq th_i} y_j = 0 \qquad \forall i \in V^+ \tag{18}$$

$$\lambda_r \geq 0 \qquad \forall r \in \Omega' \tag{19}$$

$$y_i \geq 0 \qquad \forall i \in V^+ \tag{20}$$

The objective function (16) of the LRMP maximizes the total profit in the same way as the expression (13). The constraints (17) ensure that the transporter proposes an acceptable price for at least one customer per zone in the same way as the expression (14). According to the constraints (18), if the price of the zone of customer i is set to a value lower than threshold of this customer, then customer i should be served. The binary variables y_i and λ_r are linearly relaxed to accommodate the column generation mechanism.

3.2. Pricing Problem

Let α_k and Π_i be the dual variables associated with the constraints (17) and (18), respectively.

The reduced cost of a route r is:

$$\bar{C}_r = -\sum_{i \in V^+} \gamma_r^i \cdot \Pi_i - \sum_{(i,j) \in r} c_{i,j}$$

Notice that there are exponentially many λ_r variables, one for each feasible route to be precise. However, there are only $|V^+|$ many y_i variables. In other words, in the dual of non-restricted Master Problem, there are $|\Omega|$ many dual constraints associated with variable λ_r, whereas there are $|V^+|$ many constraints that correspond to primal variable y_i ($\alpha_{z(i)} - \Pi_i \cdot l(i) \geq th_i \cdot l(i) \;\; \forall i \in V^+$). Therefore, in the dual of the Restricted Master Problem, there are fewer ($|\Omega'|$ many) constraints of the first group whilst all of the second group constraints are included. The Restricted Master Problem will generate dual values respecting the later constraints. To check the feasibility of the dual, it is sufficient to verify the reduced costs corresponding to λ_r, i.e., $\bar{C}_r$.

For any given value of $(\Pi_i)_{i\in V^+}$, if there exists a route r with a reduced cost $\bar{C}_r > \Pi_0$, it should be added into Ω'. Initially Π_0 is set to zero. During the branching phase, fleet size constraints will be added to LRMP, and Π_0 will aggregate the values of dual variables associated with these constraints. If no routes with a reduced cost strictly greater than Π_0 are found, the column generation stops and the optimal solution of the LRMP is retrieved. To be sure no such route exists, the cost of a particular elementary longest path with resource constraints should be less than or equal to Π_0.

To find such a route, the elementary longest path problem with resource constraints is defined on a directed graph $G' = (V, A')$ where the cost of any arc $(i, j) \in A'$ is $\overline{c_{i,j}} = -\frac{(\Pi_i+\Pi_j)}{2} - c_{i,j}$ and the total load of the route should be less than or equal to Q.

At first, this problem is solved heuristically. If this strategy fails, then an *ng-relaxation* is solved (Baldacci et al., 2011). In an *ng-relaxation*, each partial route r ending at node i has a forbidden set of nodes $(F(r, i))$, such that the path can be extended to a node j if, and only if, $j \notin F(r, i)$. To define the set $F(r, i)$, let s represent a position in route r, so $r(s)$ is the node in position s, $s \in \{1, ..., |r|\}$, then:

$$F(r, i) = \{r(u) : r(u) \in \bigcap_{s=u+1}^{|r|} N_{r(s)}\} \cup \{i\}$$

where $N_r(s)$ is the set of the closest "neighbors" of node $r(s)$, including itself. Due to this relation, multiple visits to a node i are allowed if there is at least one node j such that $i \notin N_j$, that is visited between the last $|N_i|$ successive visits (Pecin et al., 2017). Notice that since *ng-relaxation* will return an upper bound to the pricing problem, a node can be potentially visited several times and the resulting route is not necessarily elementary. Nevertheless these routes are added to Ω'. The non-elementary routes are eliminated during the branching and any integral solution will be feasible, therefore composed of elementary routes.

Each partial path r ending at a node i is represented by a label $L(r,i) = (\overline{C(r,i)}, load(r,i), F(r,i))$ where $\overline{C(r,i)}$ is the reduced cost, $load(r,i)$ is the total load, and $F(r,i)$ is the forbidden set of nodes. A label $L(r,i)$ is said to dominate another label $L(r',i)$ if the following conditions are satisfied :

1. $\overline{C(r,i)} \geq \overline{C(r',i)}$
2. $load(r,i) \leq load(r',i)$
3. $F(r,i) \subseteq F(r',i)$

Therefore, any extension from the node i to the depot that is feasible for r' is feasible for r, and the reduced cost of the subsequent route will be higher.

While solving the pricing problem heuristically, only elementary routes are generated and on each node, only a limited number of best labels in terms of reduced cost are kept. A relaxed dominance rule where the last condition is replaced by $|r| \leq |r'|$ is employed.

3.3. Branching Strategy

In the branch-and-price tree, the leaf with the greatest upper bound is found. If the optimal solution it found for the LRMP is not integer, three types of branching decisions are made.

1. If the total number of vehicles $\alpha = \sum_{r\in\Omega} \lambda_r$ is not integer, two branches

are created by adding these respective constraints:

$$\sum_{r\in\Omega} \lambda_r \leq \lfloor \alpha \rfloor \tag{21}$$

$$\sum_{r\in\Omega} \lambda_r \geq \lceil \alpha \rceil \tag{22}$$

It is important to remark that there may be several fleet size constraints at any leaf of the branch-and-price tree. Π_0 aggregates the dual values of all of them for the pricing problem. These constraints have no effect on the pricing problem, other than the value of Π_0. However, in practice, when the fleet size is bounded by the constraints (21), Π_0 becomes greater and pricing problem takes longer to converge.

2. If there are any non-binary price-threshold variables, we branch on the most fractional one $y_{i'}$ by adding these constraints:

$$y_{i'} \geq 1 \tag{23}$$

$$y_{i'} \leq 0 \tag{24}$$

On the branch created by constraint (23), no customer j in the same geographical area as i' such that $th_j < th_{i'}$ will be visited. Thus the graph can be reduced and for all $i \in V$, all arcs (i,j) and $(j,i) \in A'$: $z(j) = z(i')$ and $th_j < th_{i'}$ can be removed from the graph. As the routing variables λ_r do not intervene in these constraints, the structure of the reduced cost therefore the structure of the pricing problem does not change. Since the subsequent graph becomes smaller, the pricing problem converges faster.

3. If the fleet size and all price-threshold variables are integer but the final solution remains fractional, the arcs with fractional flow are sought.

The arc (i, j) with the maximum value of total flow ($\sum_{r \in \Omega' : (i,j) \in r} \lambda_r$) is chosen to be branched on. In the two created branches, either the arc (i, j) is forbidden and simply removed, or it is forced by removing all outgoing arcs from i and all incoming arcs to j from A', except (i, j). This branching schema is a version of the branching rule proposed by (Ryan and Foster, 1981) and it is based on two basic observations:

- In the sub-graph supported by a solution where the flow on every arc is either 0 or 1, the degree of each node is equal to 2 (with one incoming and one outgoing arc). In our problem, this sub-graph does not necessarily contain every node, since some of the customers reject the price therefore they are not visited. This sub-graph can be partitioned into a set of disjoint routes (Feillet, 2010).
- No identical columns can exist simultaneously in the simplex tableau of the set partitioning problem (Sarac et al., 2006).

Therefore, if all the arcs are removed or forced by this rule, in the final solution all λ_r, $\forall r \in \Omega'$ should take the value either 0 or 1.

As no constraints are added to or modified in LRMP, neither the dual nor the reduced cost of a route are affected. Therefore, the general structure of the pricing problem is intact. On the other hand, as a result of successive branching, the subsequent graph becomes smaller and less dense, and the convergence of the pricing accelerates.

At the end of this stage, if no more arcs with a fractional value are found, the base solution contains only elementary routes. A non-elementary route would contain at least one sub-tour, and at least one customer would be visited several times. Let j be the first customer of the sub-tour. Then j would have at least two different incoming arcs (i, j) and (i', j). The value of these two arcs would be fractional since the value of a non-elementary route should be strictly less than 1 due the constraints (18). This would be a contradiction with the initial hypothesis that there are no more arcs with fractional values.

4. Experimental Study

The above-mentioned method is implemented in C++ and the commercial solver CPLEX 12.9 is used to solve the LRMP and the mathematical model presented in section 2.2. All the tests are run on a PC with Intel Xeon Gold 6132 processor at 2.6 GHz and 128 GB RAM with Linux (CentOS v.6.10).

4.1. Data generation

Since the VRP-ZP is defined for the first time in this work, no instances are available. For this study, we create a group of data sets derived from 14 classical CVRP instances of Christofides et al. (1979) with no route length restriction, which are available in the repository section[1]. In each data set, p geographical zones are considered and the customers are assigned to the zones depending on their relative distance from the depot. All $i \in V^+$ are assigned to a zone V_k where $k \in \{1, ..., p\}$ as follows:

[1] `http://di.uniovi.es/iscop`

$$
\begin{aligned}
c_{0,i} \leq \frac{c_{max}}{p} &\implies i \in V_1 \\
\frac{(k-1)c_{max}}{p} < c_{0,i} \leq k \times \frac{c_{max}}{p} &\implies i \in V_k, \quad \forall k \in \{2, ..., p-1\} \\
(p-1) \times \frac{c_{max}}{p} < c_{0,i} &\implies i \in V_p \\
\text{where} \quad c_{max} = \max_{i \in V^+} \{c_{o,i}\}
\end{aligned}
$$

Three main sets are designed. For the first data set, we take three random samples of 35 customers from each of the CVRP instances and assign them in $p = 3$ zones, which yields 42 instances. Similarly, we generate larger instances by taking 3 random samples of 50 customers from each CVRP instance, with the exception of instances CMT1 and CMT6, which already have 50 customers. Therefore, we obtain 38 larger instances. For the last data set, we repeat the same process with 50 customers but assigning them to $p = 5$ zones instead of 3.

Table 1 presents average number of customers in each zone as well as the standard deviation per data set.

Table 1: Average number of customers and standard deviations per data set

Data sets		Zone 1	Zone 2	Zone 3	Zone 4	Zone 5
5 zones 50 cust.	Mean	4.45	12.20	15.39	11.74	6.21
	Standard Dev.	4.40	6.39	4.61	3.03	4.45
3 zones 50 cust.	Mean	11.87	25.28	12.85		
	Standard Dev.	3.85	7.25	5.38		
3 zones 35 cust.	Mean	7.87	17.40	9.73		
	Standard Dev.	2.59	5.13	3.94		

In addition, four different variants are created out of each of the previous instances by assigning different threshold profiles: *low*, *medium*, *high* and *random*. For each customer, the threshold is expressed as a function of distance of this customer from the depot ($th_i = \beta_i \times c_{0,i}$) that depends on the threshold profile:

- *Low threshold* corresponds to the case where customers are not willing to pay much for a home delivery ($\beta_i \in [0.8, 1.0], \ \forall i \in V^+$)
- *Medium threshold* corresponds to the case where the customers are willing to pay a price slightly above the cost of a direct trajectory ($\beta_i \in [1.0, 1.4], \ \forall i \in V^+$).
- *High threshold* corresponds to the case where the customers can pay significantly more for a home delivery ($\beta_i \in [1.6, 2.0], \ \forall i \in V^+$).
- *Random threshold* corresponds to the case where the customers have a mixed profile in terms of willingness to pay for home delivery ($\beta_i \in [0.8, 2.0], \ \forall i \in V^+$).

In total we have 168 small sized instances and 304 larger sized ones. As mentioned before, the customers of each geographical zone are sorted in an increasing order of their threshold values.

4.2. Numerical Results

The Mixed Integer Linear Model presented in section 2.2 is tested on the smallest instances: the data set with 3 zones and 35 customers, using CPLEX. The average results per threshold profile are presented in Table 2.

Table 2: Average results of instances with 35 customers

Threshold	nb OPT	GAP(%)	Time(s)
Low	4	28.69	944.2
Medium	3	14.41	1378.7
High	2	7.31	2325.1
Random	10	8.59	3721.7

Column *nb OPT* gives the number of optimally solved instances out of 42 in each threshold profile. In a total of 168 instances, only 19 optimal solutions could be found for the MIP in less than 10000 seconds. Column *GAP* contains the average gap between the best integer solutions and upper bounds of the instances for which no optimal solution is found within the time limit. Finally, column *Time* reports the average run time elapsed on the instances for which the optimum was found. That is, the instances for which the time limit is reached are not taken into account in the average run time. These results clearly indicate the difficulty of solving VRP-ZP with a Mixed Integer Linear Program.

In the following, we provide numerical results obtained by the branch-and-price algorithm with the instances introduced above. The algorithm is left to run until the optimal solution is found or a specific time limit restriction is met: at each leaf of the branch-and-price tree, if the total time elapsed is more than 10000 seconds, the algorithm stops. Otherwise, it continues branching as explained in the previous section.

Table 3 reports the results obtained on the instances with 35 customers grouped by threshold profile: *low*, *medium*, *high* and *random* (except one of the CMT2 instances with random threshold profile that could not be solved within the time limit). For each group, the average values of the optimal profit are shown. The gap column contains the average gap between the optimal solution and the upper bound of the total profit found on the root ($\frac{\Theta_{UB}-\Theta^*}{\Theta_{UB}} \times 100$). It measures the tightness of the linear relaxation of the Master Problem. The average of the total revenue of the optimal solution, number of visited customers and number of vehicles used are also presented. Finally, the branch-and-price tree size and the elapsed CPU time in seconds are shown as performance indicators.

Table 3: Average results of instances with 35 customers

Threshold	Profit	GAP(%)	Revenue	#Cust.	#Veh.	Tree size	Time(s)
Low	198.05	4.23	569.52	28.22	2.89	86.22	1767.78
Medium	382.21	2.47	761.30	28.92	2.92	107.61	1698.69
High	779.84	1.51	1174.23	30.64	3.22	59.83	1604.04
Random	436.29	1.35	787.28	26.00	2.60	34.89	449.07

Instances derived from the classical instances CMT11 and CMT13 present a very different behavior, and therefore, they are not included in the table. The main difference on these data sets is the location of the depot. In all data sets but these two, the depot is approximately in the center of the map, while on these two instances the depot is situated at the far left side. Their average results are presented in Table 4, with the exception of one of the CMT11 instances with *low* threshold value and one of the CMT11 instances with *random* threshold values, which could not be solved within the time limit.

The results in Table 3 show that as the threshold values increase from low to high, total profit, number of visited customers and fleet size increase as well. This is very clear in the case of total profit, which almost doubles every time that the thresholds increase. It is interesting however, that the increase in the number of served customers and fleet size is not as obvious as the increase in profit, especially when the threshold profile changes from *low* to *medium*. This is no surprise if we take into account that even if the number of visited customers remains constant, higher thresholds allow for higher prices and therefore higher profit. Regarding the routing costs, that is, the difference between the revenue and the profit, they also increase with the threshold value which is consistent with the increase in the number of visited customers. In the case of random thresholds, although the revenue is similar to the medium case, the overall profit is higher since fewer customers are visited using fewer vehicles. In other words, when the threshold values have a large spread, it is possible to focus on visiting customers with higher thresholds and thus reducing cost and increasing profit. Finally, we can observe that the gap between the upper bound computed at the root node and the optimal solution is below 5% for all threshold profiles, which shows that the relaxation at the root node provides a good starting point.

Table 4: Average results of instances derived from CMT11 and CMT13 with 35 customers

Threshold	Profit	GAP(%)	Revenue	#Cust.	#Veh.	Tree size	Time(s)
Low	966.79	3.78	1439.29	31.00	2.00	2.60	37895.18
Medium	1373.56	1.37	1831.27	32.00	2.00	1.80	18742.74
High	2435.53	1.34	2909.89	32.60	2.00	2.20	14965.79
Random	1231.70	1.93	1671.69	28.25	2.00	2.50	11514.87

In Table 4, the total profit and the number of visited customers increase significantly in comparison with Table 3. Since in CMT11 and CMT13, the depot is located at far left side, the customers are situated in a half circle and somewhat more clustered, hence the increase in the number of visited customers. The fleet size is the same for all threshold profiles, whereas the number of visited customers increase with the threshold value as it happened in the previous table. In the case of random threshold, once again the number of visited customers is lower than the others. It is important to underline that, these solutions are obtained at the end of the time limit and therefore sub-optimal. The gaps between the best feasible solution and the upper bound are provided.

Table 5: Average results of instances with 50 customers

Thres.	Zone	Profit	GAP(%)	Revenue	#Cust	#Veh	Tree size	Time(s)
Low	3	344.51	4.28	820.18	42.56	4.09	120.94	3854.04
Med.	3	599.04	2.49	1063.98	41.09	3.97	74.88	2597.93
High	3	1164.08	1.48	1652.98	43.75	4.38	35.63	2171.16
Rand.	3	681.00	2.15	1110.26	36.44	3.56	74.38	2519.21
Low	5	421.44	3.43	908.51	43.91	4.34	86.19	2796.63
Med.	5	690.62	2.02	1182.84	44.61	4.48	87.52	2844.09
High	5	1358.51	1.27	1866.24	46.63	4.75	36.19	3114.76
Rand.	5	757.28	1.73	1195.78	38.32	3.84	73.77	2747.18

The results for the instances with 50 customers are reported in Table 5. As in Table 3, the instances derived from CMT11 and CMT13 are not included. Since the instance size is larger, the branch-and-price tree size and the average execution time increase as expected. However, the gap between upper bound found at the root and the optimal solution is still under 5% for all threshold values, illustrating that the relaxation at the root node provides a good starting point independently on the instance size.

Contrary to what happened in Table 3, in the case of 3 zones, the routing cost and the number of visited customers are now higher in the *low* threshold instances than in the *medium* threshold ones. Although one may consider that the rise in customer thresholds would imply visiting more customers, it might not be always the case. A customer that has the highest threshold value in their zone in the *low* threshold profile instance may have the lowest value in the *medium* threshold case and hence, may not be visited in the optimal solution. In other words, the set of visited nodes on the optimal solution of a *low* threshold instance is not a subset of the set of visited nodes on the optimal solution of the *medium* threshold profile. What remains a fact in Table 5 is that increasing the thresholds increases the total profit, as expected. Regarding the *random* threshold profile, there are no significant changes with respect to Table 3. That is, the revenue is similar to the medium threshold profile but visiting fewer customers and therefore reducing costs. This happens as well when considering 5 zones. In that case, the profit, revenue, number of vehicles and visited customers are larger than in the 3-zone case. Comparing the tree size, run time and gap values of 3 and 5 zones, we can observe that increasing the number of zones does not necessarily make the problem more difficult to solve.

Table 6 contains the results of the instances derived from CMT11 and CMT13 with 50 customers. As before, one instance with 3 zones and a *low* threshold profile is omitted due to the column generation procedure not finalizing at the root node within the time limit of 10000 seconds. When 5 zones are considered, one of the instances derived from CMT13 is also

Table 6: Average results of instances derived from CMT11 and CMT13 with 50 customers

Thres.	Zone	Profit	GAP(%)	Revenue	#Cust	#Veh	Tree size	Time(s)
Low	3	1185.07	4.37	1759.91	39.20	2.80	1.40	35456.18
Med.	3	1769.20	3.01	2364.33	43.67	3.33	1.67	39677.93
High	3	3124.39	1.70	3725.14	43.67	3.17	1.00	46331.87
Rand.	3	1829.52	1.29	2351.71	38.50	2.67	1.67	18586.40
Low	5	1389.41	2.36	1980.48	43.60	3.00	1.40	40359.14
Med.	5	1918.32	1.80	2511.04	43.20	3.00	1.40	42395.38
High	5	3.380.83	0.82	3972.74	44.20	3.00	1.00	45703.92
Rand.	5	1950.92	1.08	2443.03	38.00	2.67	1.00	30720.72

omitted for *low*, *medium*, and *high* thresholds because of the same reason.

The profit, routing cost and revenue values are higher compared to the rest of the instances with 50 customers. The much higher average run times and smaller average number of explored branch-and-price tree nodes reflect the difficulty of these instances. However, the average gap values between the best feasible solution found at the end of the time limit and the upper bound found at the root node are still less than 5% in all profiles.

5. Conclusion and Perspectives

In this paper we introduced the Vehicle Routing Problem with Zone-Based Pricing. The key feature being that customers decide whether or not they accept to pay the proposed price with respect to a given personal threshold value. Once the price is accepted, the transporter has the obligation of serving them. This problem integrates the very complex task of setting zone prices into the transportation problem. In fact, the results show that even when thresholds are high and the customers are willing to pay more, the best pricing strategy is not to meet the threshold of all customers.

We have formalized the problem and proposed an exact solution approach based on branch-and-price method. To test it, we have designed data sets with different number of customers and threshold profiles. The results have shown that the method is particularly efficient when the depot is located at the central region of the map, producing optimal solutions for all these instances in relatively short amount of time for all sizes and thresholds.

In accordance with the few recent articles addressing the pricing and routing problems jointly, we assume the relation between the price and demand behavior to be known in advance. In our case, customers accept to be served if the price proposed by the transporter is less than their threshold values. We are aware that it may not be realistic to know the exact threshold values in advance. The numerical tests indicate that, even in the simpler case where thresholds are known, the problem remains extremely difficult. Nevertheless, our future works shall include a stochastic or fuzzy modeling of customer behavior up against a price proposal.

References

H. M. Afsar and N. Labadie. Team orienteering problem with decreasing profits. *Electronic Notes in Discrete Mathematics*, 41:285–293, 2013.

A. Ahmadi-Javid and R. Ghandali. An efficient optimization procedure for designing a capacitated distribution network with price-sensitive demand. *Optimization and Engineering*, 15:801 – 817, 2014.

A. Ahmadi-Javid and P. Hoseinpour. A location-inventory-pricing model in a supply chain distribution network with price-sensitive demands and inventory-capacity constraints. *Transportation Research Part E: Logistics and Transportation Review*, 82:238 – 255, 2015.

A. Ahmadi-Javid, E. Amiri, and M. Meskar. A profit-maximization location-routing-pricing problem: A branch-and-price algorithm. *European Journal of Operational Research*, 271(3):866 – 881, 2018.

N. Aras, D. Aksen, and M.T. Tekin. Selective multi-depot vehicle routing problem with pricing. *Transportation Research Part C: Emerging Technologies*, 19(5):866 – 884, 2011.

C. Archetti, D. Feillet, A. Hertz, and M. G. Speranza. The capacitated team orienteering and profitable tour problems. *Journal of the Operational Research Society*, 60(6):831–842, 2009.

E. Balas. The prize collecting traveling salesman problem. *Networks*, 19(6): 621–636, 1989.

RR. Baldacci, A. Mingozzi, and R. Roberti. New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research*, 59(5): 1269–1283, 2011.

S. Boussier, D. Feillet, and M. Gendreau. An exact algorithm for team orienteering problems. *4OR*, 5(3):211–230, 2007.

N. Christofides, A. Mingozzi, and P. Toth. The vehicle routing problem. In N. Christofides, A. Mingozzi, P. Toth, and L. Sandi, editors, *Combinatorial optimization*, pages 315–338. Wiley, Chichester, 1979.

G. B. Dantzig and J. H. Ramser. The truck dispatching problem. *Management Science*, 6(1):80–91, 1959.

G. B. Dantzig and P. Wolfe. Decomposition principle for linear programs. *Operations Research*, 8(1):101–111, 1960.

M. Dell'Amico, F. Maffioli, and P Värbrand. On prize-collecting tours and the asymmetric travelling salesman problem. *International Transactions in Operational Research*, 2(3):297 – 308, 1995.

G. Desaulniers, F. Errico, S. Irnich, and M. Schneider. Exact algorithms for electric vehicle-routing problems with time windows. *Operations Research*, 64(6):1388–1405, 2016.

G. Erdogan and G. Laporte. The orienteering problem with variable profits. *Networks*, 61(2):104–116, 2013.

F. Etebari and N. Dabiri. A hybrid heuristic for the inventory routing problem under dynamic regional pricing. *Computers & Chemical Engineering*, 95:231 – 239, 2016.

T. Ibn Faiz, C. Vogiatzis, and Md. Noor-E-Alam. A column generation algorithm for vehicle scheduling and routing problems. *Computers & Industrial Engineering*, 130:222–236, 2019. ISSN 0360-8352.

D. Feillet. A tutorial on column generation and branch-and-price for vehicle routing problems. *4OR*, 8:407–424, 12 2010.

D. Feillet, P. Dejax, and M. Gendreau. Traveling salesman problems with profits. *Transportation Science*, 39(2):188–205, 2005.

M. Fischetti, J.J. Salazar-Gonzalez, and P. Toth. *The Generalized Traveling Salesman and Orienteering Problems*, pages 609–662. Springer US, Boston, MA, 2007. ISBN 978-0-306-48213-7.

B. L. Golden, L. Levy, and R. Vohra. The orienteering problem. *Naval Research Logistics (NRL)*, 34(3):307–318, 1987.

M.K. Jepsen, B. Petersen, S. Spoorendonk, and D. Pisinger. A branch-and-cut algorithm for the capacitated profitable tour problem. *Discrete Optimization*, 14:78 – 96, 2014.

G. Laporte and S. Martello. The selective travelling salesman problem. *Discrete Applied Mathematics*, 26(2):193 – 207, 1990.

E. Legnani, S. Cavalieri, and P. Gaiardelli. *Modelling and Measuring After-Sales Service Delivery Processes*, pages 71 – 84. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

S. Liu, S. Qin, and R. Zhang. A branch-and-price algorithm for the multi-trip multi-repairman problem with time windows. *Transportation Research Part E: Logistics and Transportation Review*, 116(C):25–41, 2018.

S-C. Liu and J-R. Chen. A heuristic method for the inventory routing and pricing problem in a supply chain. *Expert Systems with Applications*, 38 (3):1447 – 1456, 2011.

D. Pecin, A. A. Pessoa, M. Poggi, and E. Uchoa. Improved branch-cut-and-price for capacitated vehicle routing. *Mathematical Programming Computation*, 9(1):61–100, 2017.

S. Ponboon, A. G. Qureshi, and E. Taniguchi. Branch-and-price algorithm for the location-routing problem with time windows. *Transportation Research Part E: Logistics and Transportation Review*, 86(C):1–19, 2016.

L. Reyes-Rubiano, A.A. Juan, C. Bayliss, J. Panadero, J. Faulin, and P. Copado. A biased-randomized learnheuristic for solving the team orienteering problem with dynamic rewards. *Transportation Research Procedia*, 47:680 – 687, 2020. ISSN 2352-1465. 22nd EURO Working Group on Transportation Meeting, EWGT 2019, 18th – 20th September 2019, Barcelona, Spain.

D. Ryan and B. Foster. An integer programming approach to scheduling. *Computer Scheduling of Public Transport*, 1:269–, 01 1981.

A. Sarac, R. Batta, and C. M. Rump. A branch-and-price approach for operational aircraft maintenance routing. *European Journal of Operational Research*, 175(3):1850–1869, 2006. ISSN 0377-2217.

D. Tas. Electric vehicle routing with flexible time windows: a column generation solution approach. *Transportation Letters*, 13(2):97–103, 2021.

T. Tsiligirides. Heuristic methods applied to orienteering. *Journal of the Operational Research Society*, 35:797–809, 1984.

P. Vansteenwegen and A. Gunawan. *Orienteering Problems*. Springer International Publishing, 2019.

P. Vansteenwegen, W. Souffriau, and D. Van Oudheusden. The orienteering problem: A survey. *European Journal of Operational Research*, 209(1):1 – 10, 2011. ISSN 0377-2217.

Y. Yu, S. Wang, J. Wang, and M. Huang. A branch-and-price algorithm for the heterogeneous fleet green vehicle routing problem with time windows. *Transportation Research Part B: Methodological*, 122:511–527, 2019. ISSN 0191-2615.

Appendix A. Detailed numerical results

In Tables A.7 to A.10 the results of all instances with 35 customers and low/medium/high/random threshold profiles are presented respectively. Column "Revenue" shows the total collected price of the optimal solution, while column "Opt." shows the optimal profit. The gap between the optimal solution and the upper bound at the root is also reported. Columns "#Cust" and "#Veh" contain the number of visited customers in the optimal solution and the fleet size, respectively. Finally, the branch-and-price tree size is reported together with the CPU time in seconds used by the solving method. The column generation procedure does not finalize at the root node in less than 10000 seconds when the thresholds are low for instance vrpnc11_2, and when they are random for instances vrpnc02_3 and vrpnc11_3, thus these instances are not reported in the corresponding table.

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01_1	548.26	204.08	1.64%	26	3	3	189.86
vrpnc01_2	535.11	151.85	3.50%	30	3	3	191.17
vrpnc01_3	497.15	155.45	2.56%	26	3	7	43.64
vrpnc02_1	430.53	111.71	15.79%	22	3	91	2556.84
vrpnc02_2	577.78	153.31	3.33%	29	4	73	2538.74
vrpnc02_3	598.12	196.67	0.48%	32	4	3	11.37
vrpnc03_1	522.45	190.16	3.02%	26	2	95	2844.43

vrpnc03_2	662.75	299.20	0.00%	30	2	1	490.58
vrpnc03_3	525.59	197.28	4.43%	27	2	117	3177.44
vrpnc04_1	525.61	204.21	5.54%	25	2	373	3914.75
vrpnc04_2	521.13	215.69	1.40%	23	2	7	1483.78
vrpnc04_3	502.77	190.24	0.31%	27	2	3	142.14
vrpnc05_1	559.31	192.49	5.71%	30	3	203	2910.62
vrpnc05_2	495.31	167.75	1.13%	27	2	27	844.78
vrpnc05_3	531.53	164.07	4.81%	29	3	257	2158.28
vrpnc06_1	654.70	233.99	7.23%	33	4	55	2478.29
vrpnc06_2	496.50	196.00	1.48%	22	2	5	396.80
vrpnc06_3	583.20	223.08	3.68%	28	3	49	615.99
vrpnc07_1	513.66	110.27	4.33%	26	4	71	1924.13
vrpnc07_2	611.01	155.40	7.91%	30	4	115	2593.25
vrpnc07_3	611.01	183.41	5.62%	31	4	67	2349.67
vrpnc08_1	638.08	233.99	4.35%	30	3	3	611.11
vrpnc08_2	556.77	228.41	5.89%	28	3	115	4983.99
vrpnc08_3	500.04	150.62	0.00%	23	2	1	119.02
vrpnc09_1	586.48	185.43	5.63%	30	3	13	407.87
vrpnc09_2	510.53	170.47	0.00%	27	2	1	277.89
vrpnc09_3	598.56	212.66	6.60%	32	3	289	3259.81
vrpnc10_1	575.98	268.95	0.82%	27	2	3	437.28
vrpnc10_2	538.08	181.10	12.21%	29	3	171	3229.02
vrpnc10_3	619.78	218.30	2.58%	31	3	3	220.76
vrpnc11_1	1353.26	884.58	2.72%	33	2	3	52959.30
vrpnc11_3	1388.34	928.88	1.55%	32	2	3	38316.90
vrpnc12_1	651.98	207.69	8.85%	34	4	73	2413.55
vrpnc12_2	615.07	195.15	5.62%	32	3	263	3356.50
vrpnc12_3	668.85	208.51	6.38%	32	3	75	2306.82
vrpnc13_1	1387.20	905.75	4.88%	32	2	1	19351.30
vrpnc13_2	1490.34	1023.27	2.89%	30	2	3	19679.60

vrpnc13_3	1577.33	1091.47	6.87%	28	2	3	59168.80
vrpnc14_1	634.36	215.70	0.42%	29	3	41	930.06
vrpnc14_2	628.30	275.36	4.17%	28	3	361	4920.69
vrpnc14_3	676.36	281.25	4.99%	25	3	71	2309.23

Table A.7: Detailed results on instances with 35 customers and low threshold

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01_1	746.45	393.48	2.55%	27	3	161	2843.01
vrpnc01_2	733.61	341.75	0.59%	30	3	3	36.90
vrpnc01_3	745.43	376.63	1.55%	29	3	117	1061.39
vrpnc02_1	688.16	285.39	7.68%	30	4	25	2466.84
vrpnc02_2	681.97	269.01	1.16%	30	4	31	353.37
vrpnc02_3	718.46	323.47	3.92%	28	4	125	1647.83
vrpnc03_1	684.47	351.16	3.74%	26	2	185	3565.66
vrpnc03_2	923.02	521.42	3.05%	34	2	111	5968.63
vrpnc03_3	692.78	343.83	0.35%	30	2	29	3723.46
vrpnc04_1	720.20	367.32	4.63%	30	3	367	3417.52
vrpnc04_2	683.40	362.82	0.94%	26	2	3	60.43
vrpnc04_3	666.95	357.05	0.00%	27	2	1	124.02
vrpnc05_1	774.39	388.63	4.82%	32	4	153	2767.80
vrpnc05_2	651.04	316.94	1.00%	28	2	69	2269.90
vrpnc05_3	733.51	333.23	1.81%	34	3	441	3655.19
vrpnc06_1	842.14	441.31	0.67%	31	3	121	897.07
vrpnc06_2	753.39	385.67	0.00%	30	3	1	30.82
vrpnc06_3	734.64	368.80	0.73%	30	3	5	138.08
vrpnc07_1	634.61	255.06	6.55%	22	4	95	1020.41
vrpnc07_2	699.76	322.02	1.63%	23	3	13	43.82
vrpnc07_3	805.87	369.07	2.01%	32	4	117	1453.24
vrpnc08_1	784.33	391.72	2.81%	27	3	3	157.24
vrpnc08_2	757.86	410.15	3.59%	30	3	241	2305.97

vrpnc08_3	707.03	329.80	0.00%	29	2	1	3082.78
vrpnc09_1	746.32	382.38	3.76%	26	2	203	1483.91
vrpnc09_2	648.18	305.86	0.99%	25	2	5	155.15
vrpnc09_3	832.48	446.27	3.73%	30	3	45	1258.53
vrpnc10_1	722.32	432.09	3.39%	23	2	17	3584.57
vrpnc10_2	725.90	362.71	3.26%	33	3	397	2866.40
vrpnc10_3	856.96	466.87	2.32%	27	3	3	223.04
vrpnc11_1	1591.94	1169.26	0.58%	30	2	3	32057.90
vrpnc11_2	1746.77	1283.02	3.16%	29	2	3	18476.00
vrpnc11_3	1723.23	1288.19	0.20%	31	2	1	10263.50
vrpnc12_1	886.99	447.83	3.56%	33	4	201	2846.99
vrpnc12_2	811.29	408.25	1.93%	30	3	17	253.20
vrpnc12_3	860.22	401.86	2.93%	31	3	105	1807.97
vrpnc13_1	1821.71	1352.09	1.32%	35	2	1	17925.60
vrpnc13_2	2036.72	1571.49	0.89%	32	2	3	21032.80
vrpnc13_3	1982.76	1486.78	3.87%	32	2	1	12433.90
vrpnc14_1	890.41	459.70	0.00%	30	3	1	93.65
vrpnc14_2	947.19	560.59	2.62%	29	3	335	1701.43
vrpnc14_3	914.92	479.26	4.74%	29	3	127	1786.71

Table A.8: Detailed results on instances with 35 customers and medium threshold

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01_1	1099.82	740.54	0.28%	29	3	5	350.33
vrpnc01_2	1099.18	713.97	0.53%	30	3	3	16.78
vrpnc01_3	1120.66	722.60	0.00%	33	4	1	22.60
vrpnc02_1	1045.75	639.90	2.96%	31	4	19	1119.36
vrpnc02_2	1144.47	707.59	1.51%	31	4	27	202.97
vrpnc02_3	1178.52	774.61	1.16%	31	4	31	171.40
vrpnc03_1	1082.31	742.28	0.90%	27	2	21	4241.47
vrpnc03_2	1369.04	1005.49	0.40%	30	2	11	6968.16

vrpnc03_3	1037.34	704.30	0.84%	25	2	3	8424.42
vrpnc04_1	1183.91	800.30	2.19%	32	3	481	3795.29
vrpnc04_2	1087.20	729.22	2.44%	30	3	39	3297.15
vrpnc04_3	976.77	666.60	0.00%	26	2	1	33.67
vrpnc05_1	1111.40	711.27	3.24%	33	4	171	1580.42
vrpnc05_2	1076.43	711.76	0.94%	33	3	3	1112.21
vrpnc05_3	1171.00	781.19	1.09%	31	3	17	195.17
vrpnc06_1	1213.55	800.76	2.02%	30	3	71	1842.19
vrpnc06_2	1179.03	772.86	1.70%	33	4	21	187.65
vrpnc06_3	1103.14	736.75	0.00%	30	3	1	43.68
vrpnc07_1	1101.96	684.18	1.72%	29	5	31	1281.00
vrpnc07_2	1202.44	746.84	0.88%	30	4	19	127.60
vrpnc07_3	1216.03	787.20	2.22%	31	4	45	841.32
vrpnc08_1	1250.95	839.17	1.44%	30	3	11	1103.30
vrpnc08_2	1093.07	760.44	2.38%	27	3	107	1085.31
vrpnc08_3	1108.32	687.66	1.19%	34	3	21	398.31
vrpnc09_1	1188.34	788.42	1.54%	29	3	3	406.78
vrpnc09_2	1109.75	719.06	2.02%	33	3	87	5378.51
vrpnc09_3	1203.12	829.52	3.03%	28	3	49	1271.45
vrpnc10_1	1127.27	808.72	2.56%	27	2	49	1549.44
vrpnc10_2	1084.01	721.97	1.53%	32	3	121	5127.01
vrpnc10_3	1234.51	833.03	0.00%	31	3	1	48.29
vrpnc11_1	2625.38	2166.70	1.07%	31	2	3	23351.20
vrpnc11_2	2712.86	2218.43	1.72%	35	3	3	17303.30
vrpnc11_3	2997.39	2532.60	0.27%	35	2	3	8717.39
vrpnc12_1	1328.77	884.48	2.01%	34	4	221	1425.32
vrpnc12_2	1286.26	878.24	0.69%	31	3	3	210.32
vrpnc12_3	1312.60	852.60	1.97%	32	3	395	2314.27
vrpnc13_1	2769.06	2281.62	2.40%	33	2	3	9593.58
vrpnc13_2	3041.44	2569.87	1.31%	31	2	3	18439.90

vrpnc13_3	3116.19	2626.87	1.65%	33	2	1	14726.90
vrpnc14_1	1322.64	905.87	1.49%	31	3	5	172.54
vrpnc14_2	1403.76	941.62	1.55%	35	4	5	157.43
vrpnc14_3	1419.14	943.35	2.67%	34	4	55	1242.47

Table A.9: Detailed results on instances with 35 customers and high threshold

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01_1	822.89	446.45	3.33%	29	3	171	1098.46
vrpnc01_2	707.56	356.91	0.91%	25	3	3	18.07
vrpnc01_3	767.64	400.72	1.52%	26	3	7	21.90
vrpnc02_1	759.73	428.56	1.01%	23	3	7	57.63
vrpnc02_2	663.12	313.28	1.85%	22	3	3	15.46
vrpnc03_1	712.23	402.25	1.03%	26	2	13	233.73
vrpnc03_2	828.94	507.99	1.68%	20	2	31	513.95
vrpnc03_3	737.32	380.94	0.38%	29	2	3	37.64
vrpnc04_1	710.98	389.77	0.53%	25	2	29	1059.94
vrpnc04_2	869.51	495.52	0.00%	31	3	1	63.17
vrpnc04_3	762.79	415.16	4.59%	27	3	145	1582.18
vrpnc05_1	731.20	406.67	3.14%	23	2	153	1151.02
vrpnc05_2	760.27	352.89	0.68%	28	3	11	336.34
vrpnc05_3	928.09	587.15	0.00%	26	2	1	11.85
vrpnc06_1	742.83	394.18	3.83%	25	3	111	818.51
vrpnc06_2	816.75	476.19	1.17%	27	3	5	151.34
vrpnc06_3	792.89	453.83	0.96%	26	3	45	312.22
vrpnc07_1	851.08	472.76	0.78%	26	4	31	554.48
vrpnc07_2	631.50	316.59	2.69%	21	3	15	369.95
vrpnc07_3	849.10	401.36	1.78%	30	4	107	1653.68
vrpnc08_1	802.51	472.71	0.88%	25	2	3	770.77
vrpnc08_2	744.81	403.60	0.00%	27	2	1	80.59
vrpnc08_3	699.50	399.50	0.34%	24	2	1	5.6

vrpnc09_1	794.10	452.30	0.98%	26	2	5	1095.97
vrpnc09_2	697.94	409.25	0.00%	25	2	1	49.36
vrpnc09_3	673.55	328.24	0.85%	25	2	13	321.99
vrpnc10_1	711.24	356.50	1.93%	29	3	3	170.03
vrpnc10_2	833.35	474.33	1.69%	27	2	83	1177.25
vrpnc10_3	709.89	409.28	0.00%	27	2	1	85.86
vrpnc11_1	1529.37	1120.73	0.36%	26	2	3	178.27
vrpnc11_2	1926.66	1474.23	0.92%	31	2	1	21636.14
vrpnc12_1	924.79	539.58	0.4%	32	3	5	280.2
vrpnc12_2	967.90	553.26	2.67%	30	3	175	905.96
vrpnc12_3	828.67	477.85	1.2%	24	2	15	541.51
vrpnc13_1	1972.63	1545.80	2.94%	28	2	1	12267.32
vrpnc13_2	1408.74	937.81	0.79%	31	2	5	1765.93
vrpnc13_3	1776.02	1322.47	3.64%	28	2	3	31848.32
vrpnc14_1	928.24	534.05	1.75%	24	3	19	128.34
vrpnc14_2	907.15	583.30	0.45%	23	2	3	27.91
vrpnc14_3	884.66	477.27	1.53%	27	3	3	14.53

Table A.10: Detailed results on instances with 35 customers and random threshold

Tables A.11 to A.14 present results of the instances with 50 customers distributed in 3 zones and different threshold profiles. For instance vrpnc13_2, derived from CMT13, the column generation procedure does not finalize at the root node in less than 10000 when thresholds are low, thus this instance is not reported in the corresponding table.

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01	764.61	296.16	5.20%	41	4	101	2805.05
vrpnc02_1	858.35	328.55	1.65%	41	5	33	2569.61
vrpnc02_2	783.03	248.47	5.68%	38	5	117	2965.47
vrpnc02_3	789.77	257.43	9.73%	39	5	51	1618.86
vrpnc03_1	821.00	386.94	2.24%	49	4	21	4084.86
vrpnc03_2	845.65	356.91	2.97%	46	4	87	8353.51

vrpnc03_3	812.91	332.93	4.55%	42	4	41	1968.22
vrpnc04_1	827.19	350.89	3.87%	48	4	377	6263.46
vrpnc04_2	757.51	298.13	6.45%	45	4	219	6379.79
vrpnc04_3	733.55	323.57	4.43%	38	3	373	4029.31
vrpnc05_1	789.78	336.47	9.00%	40	4	165	5077.48
vrpnc05_2	768.64	356.64	3.32%	40	4	137	5102.58
vrpnc05_3	803.61	423.64	1.09%	41	3	11	1664.83
vrpnc06	763.05	310.78	1.47%	39	4	91	1498.92
vrpnc07_1	904.52	334.11	9.12%	41	6	69	3024.01
vrpnc07_2	818.70	312.09	6.31%	40	5	239	3974.87
vrpnc07_3	783.85	263.52	5.83%	38	5	25	1556.57
vrpnc08_1	782.44	357.13	0.19%	39	3	3	339.61
vrpnc08_2	772.16	319.90	2.15%	47	4	3	3111.04
vrpnc08_3	791.28	297.11	9.27%	45	4	59	2274.14
vrpnc09_1	864.60	406.57	6.21%	42	3	9	3607.95
vrpnc09_2	868.51	376.57	4.66%	47	4	59	6524.54
vrpnc09_3	723.43	322.36	1.21%	41	3	3	1748.82
vrpnc10_1	689.78	292.12	2.79%	36	3	257	4250.31
vrpnc10_2	785.86	332.69	2.53%	43	4	81	4608.67
vrpnc10_3	793.27	312.39	1.59%	45	4	3	573.09
vrpnc11_1	1796.41	1194.94	3.62%	39	3	3	31839.30
vrpnc11_2	1603.59	1038.74	5.84%	35	3	1	18537.70
vrpnc11_3	1619.67	1029.81	5.84%	44	3	1	26369.10
vrpnc12_1	824.93	329.31	7.46%	44	4	57	4252.01
vrpnc12_2	1063.73	529.29	3.83%	47	5	143	6480.76
vrpnc12_3	979.14	468.88	2.08%	45	4	437	8361.16
vrpnc13_1	2104.48	1494.17	4.82%	44	3	1	75485.30
vrpnc13_3	1675.38	1167.70	1.74%	34	2	1	25049.50
vrpnc14_1	915.94	374.62	4.74%	47	5	481	5575.74
vrpnc14_2	934.49	440.10	1.12%	44	4	103	4683.28

vrpnc14_3	830.46	347.92	4.21%	44	4	17	4000.85

Table A.11: Detailed results on instances with 50 nodes, 3 zones and low threshold

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01	909.37	505.35	3.13%	31	3	105	2958.98
vrpnc02_1	1140.38	609.57	0.12%	40	5	1	60.38
vrpnc02_2	1050.98	514.68	2.26%	42	6	49	1697.17
vrpnc02_3	958.14	454.35	5.82%	36	5	19	1722.32
vrpnc03_1	1054.22	613.74	2.51%	49	4	35	3734.37
vrpnc03_2	1105.16	678.44	2.03%	36	3	29	4224.25
vrpnc03_3	1145.41	636.23	1.66%	47	4	155	4158.66
vrpnc04_1	1061.24	625.20	1.28%	41	3	31	3838.83
vrpnc04_2	1068.40	608.60	2.57%	48	4	3	2986.48
vrpnc04_3	984.36	595.53	2.03%	40	3	3	1211.45
vrpnc05_1	988.60	516.86	5.39%	41	4	225	3741.39
vrpnc05_2	1075.00	639.74	1.24%	45	4	19	1048.78
vrpnc05_3	1033.28	661.34	0.59%	40	3	3	2909.70
vrpnc06	1020.54	556.60	2.32%	39	4	105	2898.48
vrpnc07_1	1124.30	537.88	7.18%	42	6	35	1517.39
vrpnc07_2	1121.35	567.13	5.26%	44	6	73	1850.08
vrpnc07_3	954.00	477.40	4.05%	33	5	65	1890.62
vrpnc08_1	1007.83	563.82	2.83%	43	3	89	4078.95
vrpnc08_2	1050.61	605.81	1.10%	44	4	3	1970.91
vrpnc08_3	989.13	546.13	2.48%	39	3	55	2554.86
vrpnc09_1	1145.75	647.37	4.39%	48	4	213	9042.35
vrpnc09_2	1085.69	644.18	2.48%	38	3	497	6248.84
vrpnc09_3	967.77	560.26	1.13%	43	3	17	2502.78
vrpnc10_1	867.77	495.56	1.81%	30	3	15	633.95
vrpnc10_2	978.10	513.25	2.71%	44	4	25	2122.59
vrpnc10_3	1003.77	553.35	1.29%	43	4	3	1273.37

vrpnc11_1	2305.27	1679.15	3.57%	42	4	3	26712.80
vrpnc11_2	2145.83	1587.21	1.59%	41	3	3	23205.50
vrpnc11_3	2147.84	1531.69	4.87%	45	3	1	14222.50
vrpnc12_1	1118.43	632.40	1.46%	43	4	73	1344.65
vrpnc12_2	1274.86	783.33	1.94%	40	4	13	1347.97
vrpnc12_3	1205.21	720.22	0.00%	44	4	1	696.89
vrpnc13_1	2672.72	2071.34	2.92%	46	3	1	60475.50
vrpnc13_2	2713.13	2090.93	3.34%	45	4	1	92381.80
vrpnc13_3	2201.19	1654.89	1.75%	43	3	1	21069.50
vrpnc14_1	1156.73	683.29	3.00%	34	4	371	3798.26
vrpnc14_2	1268.62	790.64	0.02%	43	4	1	367.87
vrpnc14_3	1132.31	630.98	3.62%	45	4	65	2700.19

Table A.12: Detailed results on instances with 50 nodes, 3 zones and medium threshold

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01	1628.11	1144.88	1.05%	42	4	203	3223.96
vrpnc02_1	1737.74	1158.69	1.59%	46	6	45	2163.93
vrpnc02_2	1653.87	1073.89	2.16%	42	6	21	1135.80
vrpnc02_3	1595.69	1048.35	2.78%	40	5	21	1125.73
vrpnc03_1	1570.63	1148.60	1.45%	45	4	11	4783.97
vrpnc03_2	1714.49	1228.02	0.74%	44	4	3	864.46
vrpnc03_3	1674.26	1198.05	0.81%	44	4	3	1595.71
vrpnc04_1	1558.75	1100.84	1.19%	44	4	41	3074.34
vrpnc04_2	1541.53	1080.61	1.29%	47	4	3	5482.99
vrpnc04_3	1604.66	1143.46	1.23%	46	4	29	3790.55
vrpnc05_1	1556.65	1083.68	2.11%	41	4	109	4107.15
vrpnc05_2	1488.53	1101.47	0.00%	38	3	1	105.71
vrpnc05_3	1551.20	1176.84	0.00%	40	3	1	139.24
vrpnc06	1609.59	1105.89	1.70%	44	5	67	1288.08
vrpnc07_1	1750.31	1141.36	2.89%	47	7	17	2186.42

vrpnc07_2	1709.81	1140.44	2.61%	42	6	19	1349.21
vrpnc07_3	1573.66	996.59	4.30%	43	6	55	1723.48
vrpnc08_1	1586.94	1118.71	1.55%	45	4	3	2735.27
vrpnc08_2	1599.00	1147.94	0.87%	45	4	3	3102.02
vrpnc08_3	1620.77	1119.40	2.61%	46	4	11	2497.35
vrpnc09_1	1694.54	1252.77	1.36%	41	3	23	2872.91
vrpnc09_2	1743.62	1255.36	2.01%	43	4	137	3219.58
vrpnc09_3	1498.99	1095.59	0.00%	43	3	1	570.89
vrpnc10_1	1403.62	977.14	1.38%	42	4	35	1117.29
vrpnc10_2	1545.89	1081.87	0.99%	45	4	99	3371.52
vrpnc10_3	1626.56	1157.80	0.97%	43	4	3	471.61
vrpnc11_1	3865.12	3265.67	1.06%	42	3	1	17874.30
vrpnc11_2	3485.87	2887.37	1.21%	45	3	1	17350.40
vrpnc11_3	3357.75	2720.49	3.19%	46	4	1	16888.30
vrpnc12_1	1679.09	1183.01	0.80%	46	4	13	1993.64
vrpnc12_2	1994.38	1459.55	1.53%	47	5	45	3279.61
vrpnc12_3	1982.86	1445.96	1.66%	47	5	3	606.03
vrpnc13_1	4186.09	3588.82	1.84%	43	3	1	79960.00
vrpnc13_2	4130.05	3512.95	1.69%	43	3	1	133526.00
vrpnc13_3	3325.94	2771.02	1.21%	43	3	1	12392.20
vrpnc14_1	1855.91	1316.67	1.86%	44	5	81	2837.83
vrpnc14_2	1793.51	1299.76	0.56%	45	4	3	283.41
vrpnc14_3	1750.08	1267.49	0.86%	43	4	31	2377.32

Table A.13: Detailed results on instances with 50 nodes, 3 zones and high threshold

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01	945.455	567.451	2.44%	31	3	15	904.13
vrpnc02_1	1049.22	651.075	1.79%	31	4	49	1476.61
vrpnc02_2	963.829	552.337	4.13%	26	4	47	1246.36
vrpnc02_3	1118.1	633.86	1.43%	39	5	73	1696.73

vrpnc03_1	1165.45	728.528	2.59%	39	3	17	4446.29
vrpnc03_2	1335.17	859.319	1.80%	42	3	7	7962.48
vrpnc03_3	1062.88	634.574	2.24%	36	3	17	3788.78
vrpnc04_1	1170.15	733.58	1.00%	39	3	5	6963.56
vrpnc04_2	881.527	568.391	4.38%	27	2	3	10582.12
vrpnc04_3	1003.68	593.537	0.69%	38	3	117	2154.21
vrpnc05_1	972.445	588.737	1.32%	38	3	15	3560.11
vrpnc05_2	979.068	589.815	2.77%	35	3	187	6259.77
vrpnc05_3	1086.08	701.774	1.63%	36	3	37	1503.04
vrpnc06	989.306	532.401	4.08%	40	4	51	760.00
vrpnc07_1	1168.73	692.927	3.38%	38	5	57	1705.48
vrpnc07_2	1041.26	579.74	1.95%	34	5	83	431.93
vrpnc07_3	1054.89	614.465	3.23%	32	5	81	1176.27
vrpnc08_1	1062.98	628.355	3.61%	34	3	53	2421.42
vrpnc08_2	1032.11	604.037	2.41%	36	3	201	2451.24
vrpnc08_3	1010.86	614.079	1.51%	39	3	7	1099.81
vrpnc09_1	1099.54	655.519	3.09%	36	4	199	1945.26
vrpnc09_2	1057.87	621.55	0.81%	42	3	49	3337.87
vrpnc09_3	1123.63	740.877	0.92%	33	3	5	208.42
vrpnc10_1	1062.66	683.73	2.09%	33	3	389	3527.59
vrpnc10_2	1077.9	736.744	1.74%	32	3	311	2228.45
vrpnc10_3	1157.41	722.116	1.95%	44	4	3	1199.47
vrpnc11_1	2653.24	2179.62	0.22%	36	2	1	29310.93
vrpnc11_2	2043.45	1516.04	2.27%	41	3	1	16912.23
vrpnc11_3	2649.43	2050.89	1.27%	44	3	1	26013.84
vrpnc12_1	1176.36	683.917	3.65%	38	4	141	1892.90
vrpnc12_2	1417.08	940.666	0.00%	42	4	1	91.39
vrpnc12_3	1434.24	949.919	1.68%	39	4	121	2270.69
vrpnc13_1	2296.98	1769.01	1.10%	39	3	1	16380.90
vrpnc13_2	2315.66	1804.62	1.09%	35	3	3	13767.20

vrpnc13_3	2151.52	1656.95	1.81%	36	2	3	9133.40
vrpnc14_1	1249.54	759.891	1.27%	36	4	9	379.85
vrpnc14_2	1293.92	818.754	0.51%	42	4	3	212.70
vrpnc14_3	1284.94	809.24	2.39%	39	4	27	729.97

Table A.14: Detailed results on instances with 50 nodes, 3 zones and random threshold

Tables A.15 to A.18 present results of the instances with 50 customers distributed in 5 zones and different threshold profiles. For the instance vrpnc13_1, derived from CMT13, the column generation procedure does not finalize at the root node in less than 10000 when thresholds are low, medium or high, thus this instance is not reported in the corresponding table. It is also the case of instance vrpnc03_3 with average threshold profile.

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01	886.92	379.26	4.75%	45	5	39	2273.19
vrpnc02_1	917.84	334.36	4.34%	46	6	17	1747.76
vrpnc02_2	934.74	361.69	6.81%	44	6	15	1949.33
vrpnc02_3	892.09	338.26	4.20%	46	6	21	1877.46
vrpnc03_1	881.84	380.65	2.99%	48	4	87	3114.07
vrpnc03_2	938.60	457.76	3.22%	46	3	23	6089.08
vrpnc03_3	833.52	376.95	4.31%	44	4	141	3651.48
vrpnc04_1	914.47	423.72	6.72%	43	4	31	5066.10
vrpnc04_2	817.05	367.60	4.70%	45	4	21	2506.23
vrpnc04_3	758.25	378.16	3.60%	39	3	237	5866.36
vrpnc05_1	847.97	420.03	2.54%	43	4	77	2819.32
vrpnc05_2	765.90	377.72	3.45%	37	3	81	2338.12
vrpnc05_3	1000.50	506.91	3.82%	46	4	141	3044.63
vrpnc06	875.09	403.65	2.18%	40	4	15	2069.36
vrpnc07_1	822.79	279.41	1.98%	43	6	27	1940.11
vrpnc07_2	872.63	307.30	2.01%	43	6	51	1684.87
vrpnc07_3	798.54	262.68	7.54%	42	6	59	1850.35
vrpnc08_1	931.53	458.84	2.19%	46	4	129	6142.59

vrpnc08_2	912.31	408.67	1.62%	45	4	17	998.57
vrpnc08_3	887.42	401.93	3.54%	46	4	121	3723.84
vrpnc09_1	861.10	391.80	2.91%	46	4	47	2537.22
vrpnc09_2	966.90	535.61	2.95%	44	4	43	2392.72
vrpnc09_3	1030.92	544.85	2.42%	46	4	71	2932.67
vrpnc10_1	856.79	379.13	6.88%	41	4	17	2562.41
vrpnc10_2	883.64	442.04	4.02%	47	5	191	4986.41
vrpnc10_3	852.31	426.43	4.86%	41	4	203	2332.11
vrpnc11_1	1981.45	1372.96	1.74%	42	3	3	29511.10
vrpnc11_2	2007.42	1434.94	2.63%	42	3	1	46867.70
vrpnc11_3	1888.80	1319.06	2.47%	40	3	1	35108.00
vrpnc12_1	1015.78	514.12	2.79%	43	4	641	4336.87
vrpnc12_2	1047.77	560.28	0.26%	44	4	7	527.21
vrpnc12_3	1114.37	598.04	4.12%	43	4	59	2420.27
vrpnc13_2	1997.25	1404.54	2.00%	46	3	1	41879.80
vrpnc13_3	2027.47	1415.53	2.96%	48	3	1	48429.10
vrpnc14_1	978.53	490.72	1.05%	47	4	3	613.37
vrpnc14_2	998.53	505.76	0.00%	41	4	1	92.51
vrpnc14_3	975.57	471.71	0.87%	45	4	125	3005.56

Table A.15: Detailed results on instances with 50 nodes, 5 zones and low threshold

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01	1117.48	641.41	1.39%	43	4	11	1227.97
vrpnc02_1	1203.85	629.09	1.52%	46	6	25	1507.70
vrpnc02_2	1227.81	647.97	2.95%	45	6	17	1521.23
vrpnc02_3	1147.17	589.86	2.37%	46	6	47	1125.64
vrpnc03_1	1182.01	686.93	2.06%	46	4	47	2369.73
vrpnc03_2	1209.97	726.74	0.60%	48	3	15	3673.79
vrpnc04_1	1186.05	699.76	3.46%	42	4	251	3446.88
vrpnc04_2	1042.52	593.07	2.40%	45	4	251	2876.03

vrpnc04_3	1070.05	635.86	1.67%	48	4	49	5986.10
vrpnc05_1	1048.38	629.04	1.46%	42	4	5	1142.46
vrpnc05_2	1032.12	602.64	2.94%	43	4	55	2267.04
vrpnc05_3	1347.21	849.10	0.79%	47	4	49	3382.90
vrpnc06	1087.86	624.79	3.05%	41	4	29	1693.15
vrpnc07_1	1147.64	543.04	3.66%	47	7	49	1377.34
vrpnc07_2	1167.77	541.71	4.49%	46	7	13	1492.71
vrpnc07_3	1058.45	506.37	1.47%	44	6	57	1677.78
vrpnc08_1	1132.48	691.13	1.29%	42	3	63	8872.39
vrpnc08_2	1171.07	703.38	1.57%	43	4	31	2891.77
vrpnc08_3	1167.29	689.82	1.64%	44	4	285	6256.96
vrpnc09_1	1116.33	655.93	1.79%	44	4	35	5260.58
vrpnc09_2	1213.62	778.33	2.21%	45	4	51	4370.18
vrpnc09_3	1264.82	809.02	1.10%	44	4	23	2509.81
vrpnc10_1	1198.74	708.23	2.52%	44	4	19	2623.47
vrpnc10_2	1133.92	693.07	1.67%	48	4	209	5187.59
vrpnc10_3	1075.81	663.66	1.02%	39	4	7	530.20
vrpnc11_1	2445.26	1841.48	1.29%	42	3	3	35346.20
vrpnc11_2	2666.22	2075.67	1.70%	43	3	1	28419.30
vrpnc11_3	2442.21	1872.61	1.61%	41	3	1	51501.30
vrpnc12_1	1270.01	777.85	1.72%	42	4	49	1786.09
vrpnc12_2	1320.65	833.81	0.00%	44	4	1	130.67
vrpnc12_3	1417.28	887.83	2.82%	46	5	551	4315.75
vrpnc13_2	2485.93	1877.24	2.54%	44	3	1	29035.00
vrpnc13_3	2515.58	1924.58	1.83%	46	3	1	67675.10
vrpnc14_1	1224.80	747.24	1.28%	45	4	5	1291.74
vrpnc14_2	1358.71	830.91	2.68%	46	5	333	2417.86
vrpnc14_3	1326.27	791.67	2.66%	48	5	81	2953.33

Table A.16: Detailed results on instances with 50 nodes, 5 zones and medium threshold

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01	1735.83	1216.65	1.48%	47	5	37	1436.72
vrpnc02_1	1883.03	1300.92	1.59%	46	6	27	1295.73
vrpnc02_2	1864.07	1236.23	2.68%	50	7	11	1412.56
vrpnc02_3	1732.48	1172.42	1.64%	46	7	83	1156.70
vrpnc03_1	1795.05	1270.64	1.16%	50	4	21	3311.59
vrpnc03_2	1887.01	1414.88	1.42%	45	3	3	13508.80
vrpnc04_1	1917.54	1382.19	2.64%	49	5	5	3400.37
vrpnc04_2	1690.19	1227.08	1.07%	49	4	13	6532.18
vrpnc04_3	1670.62	1236.93	1.04%	47	4	45	5643.26
vrpnc05_1	1704.47	1276.21	0.57%	43	4	3	2158.46
vrpnc05_2	1726.49	1284.48	1.06%	47	4	47	4622.29
vrpnc05_3	2111.45	1605.86	0.61%	48	4	15	4575.92
vrpnc06	1781.35	1267.16	0.85%	46	5	21	1452.63
vrpnc07_1	1747.07	1146.86	1.64%	47	7	23	1154.16
vrpnc07_2	1833.61	1194.98	1.94%	47	7	13	1014.64
vrpnc07_3	1666.47	1098.95	2.14%	45	7	19	1291.19
vrpnc08_1	1840.53	1366.51	0.94%	47	4	87	5709.34
vrpnc08_2	1812.13	1308.61	0.87%	44	4	5	1641.45
vrpnc08_3	1881.25	1392.34	1.17%	45	4	31	4154.06
vrpnc09_1	1793.38	1324.09	1.14%	45	4	21	2970.37
vrpnc09_2	1982.69	1526.36	1.63%	47	4	51	4645.72
vrpnc09_3	2007.23	1514.74	0.52%	48	4	3	2655.09
vrpnc10_1	1770.97	1293.17	1.17%	43	4	21	2820.73
vrpnc10_2	1718.29	1289.77	1.02%	45	4	111	8110.22
vrpnc10_3	1804.63	1324.93	1.51%	49	5	119	3623.69
vrpnc11_1	3904.13	3294.62	0.16%	45	3	1	17669.40
vrpnc11_2	4225.44	3629.89	0.91%	44	3	1	26374.40
vrpnc11_3	3708.67	3133.03	1.15%	42	3	1	82861.30
vrpnc12_1	2033.79	1542.05	0.00%	44	4	1	137.95

vrpnc12_2	2154.93	1632.98	0.94%	48	5	3	308.21
vrpnc12_3	2203.42	1666.79	1.58%	47	5	231	2193.98
vrpnc13_2	4051.38	3455.40	0.90%	45	3	1	72349.30
vrpnc13_3	3974.08	3391.20	0.96%	45	3	1	29265.20
vrpnc14_1	2026.76	1539.98	0.77%	46	4	3	1654.68
vrpnc14_2	2144.53	1612.91	0.77%	48	5	3	471.55
vrpnc14_3	2084.19	1548.29	1.42%	48	5	9	821.62

Table A.17: Detailed results on instances with 50 nodes, 5 zones and high threshold

Instance	Revenue	Opt.	GAP(%)	#Cust	#Veh	Tree size	Time(s)
vrpnc01_v	1111.19	655.03	0.69%	40	4	113	980.19
vrpnc02_1	1223.28	711.84	1.15%	42	6	23	2720.55
vrpnc02_2	998.91	586.60	1.17%	34	5	77	934.35
vrpnc02_3	1126.76	702.74	2.07%	37	5	39	2412.58
vrpnc03_1	1283.90	828.78	3.74%	38	3	19	9243.06
vrpnc03_2	1220.36	799.77	0.80%	39	3	33	5825.25
vrpnc04_1	1081.00	693.27	1.36%	37	3	9	1683.72
vrpnc04_2	1325.18	859.24	1.31%	42	4	461	5399.19
vrpnc04_3	1050.22	673.76	0.92%	34	3	11	2576.94
vrpnc05_1	971.95	601.28	5.95%	31	3	31	3756.26
vrpnc05_2	1162.66	779.49	1.28%	38	3	3	689.62
vrpnc05_3	1227.15	771.26	1.96%	40	4	311	2406.59
vrpnc06_v	1045.94	612.53	1.84%	35	4	177	1206.72
vrpnc07_1	1076.82	614.88	1.75%	35	5	37	2175.81
vrpnc07_2	1174.65	657.87	1.14%	40	5	21	2590.32
vrpnc07_3	1205.77	759.09	0.33%	36	5	3	37.63
vrpnc08_1	1161.43	772.78	0.68%	35	3	3	3331.44
vrpnc08_2	1199.81	745.56	1.04%	40	3	33	4813.20
vrpnc08_3	1196.53	761.80	0.00%	39	3	1	69.98
vrpnc09_1	1116.79	713.10	2.79%	38	3	365	5174.93

vrpnc09_2	1160.82	740.73	2.78%	45	4	19	6287.07
vrpnc09_3	1237.96	888.35	2.17%	30	2	93	2610.84
vrpnc10_1	1081.89	653.05	4.05%	41	4	71	3724.04
vrpnc10_2	1221.36	831.50	0.95%	33	3	49	3533.71
vrpnc10_3	1132.34	711.13	0.00%	43	4	1	115.52
vrpnc11_1	2353.40	1878.49	0.93%	37	3	1	22744.10
vrpnc11_2	2182.67	1683.78	1.01%	34	3	3	13854.90
vrpnc11_3	2596.47	2059.58	1.94%	41	3	1	36217.50
vrpnc12_1	1435.97	936.94	0.62%	40	4	3	113.93
vrpnc12_2	1689.50	1196.51	0.13%	45	4	5	468.43
vrpnc12_3	1371.77	887.64	0.51%	41	4	3	163.42
vrpnc13_2	2330.15	1876.91	0.96%	38	2	1	15921.30
vrpnc13_3	2721.26	2265.19	0.27%	36	2	1	33049.20
vrpnc14_1	1297.25	786.29	2.75%	46	5	33	976.58
vrpnc14_2	1298.55	767.82	3.21%	42	5	137	3116.25
vrpnc14_3	1264.82	806.35	3.10%	39	4	35	3486.75

Table A.18: Detailed results on instances with 50 nodes, 5 zones and random threshold