International Journal of Mathematical Theory and Applications
Vol 1, Issue 3 2025

Formalizations of the periodic vehicle routing problem with time-window and time-spread constraints

Amala Kizilay *, Abidinbgc

Department of Mathematics

Article Info
Received: 30-05-2025 Revised:08-07-2025 Accepted:19-07-2025 Published:29-07-2025

Abstract
The famous vehicle routing issue has an offshoot called the periodic vehicle routing problem (PVRP). The problem of high-value cargo transportation using PVRP that incorporates time windows and time spread limits (PVRP-TWTS) is tackled in this research. Over the course of many planning periods, a fleet of diversely capacitated vehicles must fulfill customers' period-specific expectations in the PVRP-TWTS. Due to security considerations, waiting time is not permitted for the cars, and client arrival times should be erratic within their time slot throughout the planning periods. For the PVRP-TWTS, this research suggests new models based on mixed-integer linear programming (MILP) and constraint programming (CP). In addition, we provide a lower limit and other valid inequalities to bolster the MILP and CP models that have been suggested. In the operations research community, CP is still less well-known than MILP, despite its usefulness for a variety of optimization issues. Utilizing CP's efficacy in resolving the PVRP-TWTS is the primary goal of this research. Based on our research, this is the only publication to date to provide a CP model for PVRP-TWTS. The issue may be better understood with a comparison of the CP and MILP models. By making adjustments to the widely-used literature benchmark set, we assess how well the suggested MILP and CP models perform. Extensive computational studies demonstrate that, when comparing solution quality, the CP model outperforms the MILP model.

Keywords: *Constraint programming Mixed integer programming Periodic vehicle routing problem Time windows Capacitated vehicles*

1. Introduction

By extending the original vehicle routing problem (VRP) to include several planning periods, the periodic vehicle routing problem (PVRP) determines the paths that vehicles take. Period by period, routes are planned for a fleet of vehicles with varying capacities, taking into account the following: the beginning and ending points of each vehicle's tour; the delivery of client needs; and the consideration of vehicle capacities. Then, across the planning horizon, the PVRP seeks to identify a collection of routes for the vehicles that minimizes the overall trip time (or distance).
This study tackles the periodic vehicle routing issue with time windows and time spread limits (PVRP-TWTS), a challenge that crops up in the delivery of high-value shipments. It is possible to model the transportation operations of high-value cargo firms as a PVRP since they often plan the vehicle routes across many days and visit most of their clients each day. In addition to the limits and assumptions already described, PVRP-TWTS also includes client time windows. Each customer may only be visited inside their designated time window. Transportation of valuable shipments also favors flexible arrival timings owing to security considerations. Customers are therefore subject to time spread limits as well. To be more specific, there must be a set time constant variation between any two visits to the same client within the planning periods in terms of arrival times. Furthermore, vehicle waiting periods are not permitted for security reasons.
In practice, the PVRP-TWTS is used when the daily routes for the ATM loading and unloading activities are planned. The

Money, diamonds, and other valuables may be securely transported from one place to

another with the help of Cash in Transit (CIT) firms. In addition to banks, these CIT firms also have consumers who buy at supermarkets and other retail establishments.

2. Literature review

Many exact and heuristic methods have been proposed in the literature for vehicle routing problems such as [9,10]. The VRP has been the subject of three recent extensive assessments, as shown in [11–13]. Moreover, [14] provides a survey on PVRP that was conducted recently.
Order dissimilarity and arrival time dissimilarity are the two primary methods for building unexpected routes in the literature. Dissimilarities in order are

often handled as issues with peripatetic vehicle routing. The objective of the m-peripatetic vehicle routing problem (m-PVRP) is to find a collection of cost-minimizing edge-disjoint routes over m periods that visit each client exactly once and use the edge between any two customers no more than once throughout those periods. Several upper and lower limits for the m-PVRP issue were suggested in the original study by [15]. The same authors went on to provide an accurate solution for the m-PVRP based on polyhedral and column generating procedures in [16], in addition to another lower bounding strategy. One strict requirement of the m-PVRP is that, throughout the u periods, no edge must be visited twice. This restriction was later relaxed in [17] with the introduction of a similar problem called the k-dissimilar VRP. In this problem, the goal is to find k different VRP solutions with a similarity level below a certain threshold. It should be noted that the k-dissimilar VRP does not expressly prohibit the repeated usages of an edge.
According to Akgün et al. [18], pathways that are geographically distributed are not always the outcome of edge dissimilarity. Using a multi-criteria shortest path method, the authors of [19] created physically distinct routes by analyzing the junction zones of those routes. Then, they used those pathways to find the Pareto-optimal ones. Minimizing transportation costs and security risks while transporting high-value commodities was achieved using a bi-objective function in [20]. A weighted total of a route's danger and its resemblance to other routes is the combined risk criteria that the authors came up with. Additionally, a risk metric connected to the amount of loaded money and the distance traveled by the vehicles was created by Talarico et al. [21,22].
Changing the visiting (coming) times of each client over the planning periods is another tactic to build unpredictable pathways. The goal of studying visiting time diversification in the context of nightly safety guarding is to produce a collection of routes that are distinct with respect to visiting time [23]. Take note of how they took it as read that every client had the same timing window.
With the goal of minimizing overall driving time and keeping a similarity measure within a specific threshold, Constantino et al. [24] examined the challenge of collecting money from parking meters. By formulating the similarity measure of two tours as the proportion of tasks that are served in the same time, they were able to build a MILP model and a metaheuristic for the issue. Yan et al. [25] investigated arrival time diversification as well, supposing that a customer's visit time must differ by a minimum of βp minutes from their most recent visit time. They reframed it as a network flow issue involving integers and several commodities.
Two research [1,2] that take PVRP identical to our PVRP-TWTS into consideration are as far as we are aware. The PVRP with arrival time dissimilarity was modeled in the initial research by Michallet et al. [1] by differentiating the arrival times at each client by at least a set amount of time.

maintained at a steady rate throughout the planning stages. Customers are also given time windows in the study of [1], and cars are not allowed to wait. They came up with a time-dependent penalty function to punish unrealistic solutions that breach the arrival time dissimilarity and the time window constraints in their iterative local search for the issue.
Subsequently, Hoogeboom and Dullaert [2] investigated an issue comparable to that of [1], in which a distinct time spread constant is specified for every client. Rather of using the periodic setting of [1], a rolling horizon method is used in [2]. Their rolling horizon method involves solving a VRP using many time windows every period, after which the solution area is refreshed by removing arrival time slots from prior periods. Along with four penalization strategies to enable impractical answers via the heuristic technique, they suggested an iterative granular tabu search algorithm to solve the issue.
Please be aware that the two studies cited above ([1,2]) make the underlying assumptions that consumer demand is constant over time and that all cars are of equal capacity. On the other hand, in practice, CIT firms often operate with a diverse fleet of cars, and client demand fluctuates throughout the year for obvious seasonal reasons. As a result, to make it more reflective of reality, we expand the PVRP-TWTS of [1] to account for diverse vehicles with varying capacity and clients with period-specific requests. To address this deficiency, this research presents new precise solution approaches based on MILP and CP models for this PVRP-TWTS extension that is better grounded in reality.
This research also intends to highlight the influence and growing appeal of CP techniques by demonstrating their efficacy on such a complicated PVRP-TWTS. Lastly, we hope that other researchers and specialists in the area will find our work beneficial as a starting point for their own CP and PVRP research experiments.

3. Problem definition

In this paper, we consider the PVRP-TWTS for the ATM loading process of the banks. Generally, banks plan the visits of the vehicles to the customers (ATMs) over several days, where the customers must be visited within their time windows each day. Namely, in each day (period), they assign customers to the vehicles, and, then, they plan the routes of the vehicles regarding these assignments. In the ATM loading process, a team of workers departs from the depot, loads the ATMs, which are on their path, and turns back to the depot again. Note that the workers should load each ATM within specified periods, i.e., time windows, preferably outside the working hours of the customers. Furthermore, since each vehicle has an individual capacity, the total demand of the customers on the vehicle's route should not exceed the capacity of the vehicle. As the demand of the customers (ATMs) generally varies from day to day, each customer has a different demand in each period.

Since the vehicles are carrying valuable goods, i.e., money for the ATMs, arrivals earlier than the start of the customer's time window are forbidden to ensure the security of the vehicles. Furthermore, the arrival times of the vehicles to a customer should be irregular over the planning periods. Namely, the arrival times of any two visits to the same customer must vary by at least a certain amount of time ϵ over the periods, without violating the time window of the customer. In this way, the arrival times to a customer (ATM) vary between the periods. Since it is going to be hard to track the vehicles, the exact time of the visits to the ATMs would not be known. Consequently, the goal of the banks is to provide the security of transfers over several planning periods with minimum travel time. Note that this problem with additional limitations, i.e., the PVRP-TWTS, can also be seen in other high-value shipment transportation processes of the CIT companies.

The PVRP-TWTS can be formally defined on a directed graph with a planning horizon P of several periods, i.e., days. The set of nodes $N = \{0, 1, \dots, n,$
$n + 1\}$ contains the set of customers, where nodes 0 and $n + 1$ denote the depot. There is a set V of heterogeneous vehicles with different capacities, where each vehicle k has an individual capacity W_k. The service at each customer i includes providing an amount $d_{i,p}$ of goods in a period $p \in P$, with a service time s_i, without arriving earlier than a_i and not after the
b_i. Note that waiting at a customer until the beginning of its time window is forbidden. Furthermore, for any two periods, p and p', a minimum time constant ϵ should split the arrival times at a customer, corresponding to time spread constraints.

The PVRP-TWTS aims to construct a set of vehicle routes in each period that minimizes total travel time while distinguishing the arrival times. In a route, a vehicle departs from the depot with a load that complies with its capacity; visits a subgroup of customers, which are assigned to it, within their time windows; delivers to these customers their demands; and returns to the depot. Note that the PVRP-TWTS considers both the assignment of the vehicles to customers and the routing of the vehicles. The PVRP-TWTS is NP-hard, since the basic single-period version of the problem, i.e., the vehicle routing problem, is known to be NP-hard [26].

A variant of the PVRP-TWTS was initially studied in the literature by [1], and a MILP formulation was provided for the problem. However, the authors [1] assume that all vehicles are identical, and the demand of a customer is the same in all periods. In this paper, we consider heterogeneous vehicles with different capacities and customers with period-specific demands to reflect the real practice. Consequently, we extend the

Objective function

$$Minimize\ Z = \sum_{p\in P} \sum_{k\in V} \sum_{i,j\in N|i\neq j} t_{i,j} x_{i,j,k,p}$$

Constraints

(1)

MILP model of [1] by including heterogeneous vehicles and customers with period-specific demands. Furthermore, we propose a novel CP model for the studied PVRP-TWTS as well as valid inequalities and a lower bound.

4. Mathematical models

As mentioned in Section 3, this paper formulates a MILP model and a CP model for the PVRP-TWTS using the OPL modeling language [27]. All parameters and sets of the PVRP-TWTS used in the proposed MILP and CP models are presented as follows.

Sets & parameters

N: Set of nodes = {0, 1,…, n, n+1}
C: Set of customers = {1,…, n}
V: Set of vehicles
P: Set of periods
W_k: Capacity of vehicle $k \in V$

$t_{i,j}$: Travel time from node $i \in N$ to node $j \in N$

$d_{i,p}$: Demand of node $i \in N$ at period $p \in P$. Note that the demand of the depot nodes equal to 0 in all periods.

a_i: Start time of the time window of node $i \in N$

b_i: End time of the time window of node $i \in N$

s_i: Service time at node $i \in N$. Note that the service time of the depot nodes equal to 0.

ϵ: Minimum time-spread constant between any two arrivals at a customer such that:
$\epsilon \leq \epsilon_{up} = \min_{i \in C}\{b_i - a_i\}/(|P| - 1)$

M: A sufficiently large integer:
$M = \max_{i \in C}\{b_i - a_i\} + \epsilon$

B: A sufficiently large integer:
$B = \max_{i \in N}\{b_i\} + \epsilon$

4.1. Mixed integer linear programming (MILP) model

The decision variables of the MILP model are defined as follows:

Decision variables

$$x_{i,j,k,p} = \begin{cases} 1, \text{if node } i \text{ is assigned to vehicle } k \\ 0, \text{otherwise} \end{cases}$$

$$\sum_{k \in V} y_{i,k,p} = 1 \quad \forall i \in C, p \in P \tag{2}$$

$$\sum_{k \in V} y_{0,k,p} \leq |V| \quad \forall p \in P \tag{3}$$

$$\sum_{i \in N | i \neq j} x_{i,j,k,p} = y_{j,k,p} \quad \forall j \in N \backslash \{0\}, k \in V, p \in P \tag{4}$$

$$\sum_{j \in N | i \neq j} x_{i,j,k,p} = y_{i,k,p} \quad \forall i \in N \backslash \{n+1\}, k \in V, p \in P \tag{5}$$

$$\sum_{i \in N \backslash \{n+1\}} x_{i,n+1,k,p} = 1 \quad \forall k \in V, p \in P \tag{6}$$

$$\sum_{i \in N} d_{i,p} * y_{i,k,p} \leq W_k \quad \forall k \in V, p \in P \tag{7}$$

$$ts_{i,k,p} + s_i + t_{i,j} - (b_i + s_i + t_{i,j})(1 - x_{i,j,k,p}) \leq ts_{j,k,p} \quad \forall i, j \in N | i \neq j, k \in V, p \in P \tag{8}$$

$$ts_{i,k,p} + s_i + t_{i,j} + b_j(1 - x_{i,j,k,p}) \geq ts_{j,k,p} \quad \forall i, j \in N | i \neq j, k \in V, p \in P \tag{9}$$

$$ts_{i,k,p} \geq a_i \quad \forall i \in N, k \in V, p \in P \tag{10}$$

$$ts_{i,k,p} \leq b_i \quad \forall i \in N, k \in V, p \in P \tag{11}$$

$$ts_{i,k,p} - ts_{i,k',p'} + M(1 - z_{i,p,p',k,k'}) \geq \epsilon \quad \forall i \in C, k, k' \in V, p, p' \in P | p \neq p' \tag{12}$$

$$ts_{i,k',p'} - ts_{i,k,p} + M * z_{i,p,p',k,k'} \geq \epsilon \quad \forall i \in C, k, k' \in V, p, p' \in P | p \neq p' \tag{13}$$

The objective function of the PVRP-TWTS (Eq.1) minimizes the total travel time of the vehicles. The model aims to reduce the total cost of the routes within the planning horizon. The constraint (Eq.2) ensures that, in each period, each customer is visited precisely once by a single vehicle, while the constraint (Eq.3) guarantees that the number of departures from the depot is not greater than the total number of vehicles in each period. Constraints (Eq.4- Eq.5) state that there must be arrival to each node except the depot node 0, and there must be a departure from each node except the depot node $n + 1$, respectively. All vehicles should return to the depot node $n + 1$, specified by the constraint (Eq.6).

Since vehicles have different capacities, the customer-vehicle assignments should respect those vehicle capacities. The constraint (Eq.7) guarantees that the total demand assigned to a vehicle does not exceed its capacity.

The first valid inequality (Eq.14) is written concerning this assumption. Note that the latest time for a vehicle to arrive at a customer is the upper limit of its time window. If a vehicle visits a customer i at the upper limit of its time window (b_i) and serves it with a service time s_i; after that, travels to another customer j with travel time ($t_{i,j}$), and arrives at the customer j earlier than the lower limit of its time window (a_j), then the vehicle should wait to serve the customer j until its time window is opened. Since the waiting times are not allowed, the vehicle cannot visit the customer j immediately after visiting the customer i if the aforementioned situation occurs.

The second inequality (Eq.15) is written concerning the time window restrictions of the nodes. Note that the earliest time for a vehicle to serve a customer is the lower limit of its time window. If a vehicle visits a customer i at the lower limit of its time window (a_i) and serves it with a service time s_i; after that, travels to another customer

j with travel time ($t_{i,j}$), and arrives at
the customer j later than the upper limit of its time window (b_j), then, the vehicle cannot visit the customer j immediately after visiting customer i, as the upper limit of the time window of customer j is exceeded.

4.2. Constraint programming (CP) model

The following parts present the decision variables, the objective function, and the constraints for the CP model. In the CP model, optional interval variables are defined to represent serving node i with a service time s_i regarding its time window by vehicle k at period p. Furthermore, the sequence variables are defined to represent the route of a vehicle k at period p. Each sequence constraint collects all the optional interval variables related to a certain vehicle.

Decision variables

$x_{i,k,p}$: Optional interval variable for visiting the node i by vehicle k in period p with a service time s_i, which is defined in a domain of $[a_i, b_i + s_i]$.
$vh_{k,p}$: Sequence variable of vehicle k for period p over $\{x_{i,k,p} | i \in N\}$.

Objective function

$$\sum_{i \in N} d_{i,p} * presence(x_{i,k,p}) \leq W_k \quad \forall k \in V, p \in P \tag{21}$$

$$first(vh_{k,p}, x_{0,k,p}) \quad \forall k \in V, p \in P \tag{22}$$

$$last(vh_{k,p}, x_{n+1,k,p}) \quad \forall k \in V, p \in P \tag{23}$$

$$noOverlap(vh_{k,p}, t_{i,j}, 1) \quad \forall k \in V, p \in P \tag{24}$$

$$abs\,(start(x_{i,k,p}, B + \epsilon) - start(x_{i,k',p'}, B)) \geq \epsilon \quad \forall i \in C, k, k' \in V, p, p' \in P | p \neq p' \tag{25}$$

$$start(x_{n+1,k,p}) - end(x_{0,k,p}) = \sum_{i \in N \setminus \{n+1\}} t_{i,Next(vh_{k,p}, x_{i,k,p}, i, i)} + \sum_{i \in C} s_i * presence(x_{i,k,p}) \quad \forall k \in V, p \in P \tag{26}$$

In the objective (Eq.16), the CP model minimizes the total travel time of the vehicles over all periods. In the equation (Eq.16), the *next* operator displays the next node in the route of the vehicle k, immediately after visiting node i in period p. The constraint (Eq.17) ensures that each customer is visited by only a single vehicle at each period, and the constraint (Eq.18) guarantees that each customer is visited exactly $|P|$ times over the planning horizon. Constraints (Eq.19-Eq.20) ensure that the vehicles start and end their routes at the depot in each period. Note that nodes 0 and $n + 1$ represent the depot. Since a vehicle visits a subset of customers at each period, the total demand of those customers should not exceed the capacity of the vehicle, which is assured by the constraint (Eq.21).

According to the constraints (Eq.22, Eq.23), the depot should be the first and the last node in the route of a vehicle for each period. A vehicle cannot visit more than one node at the same time. Thus, the constraint (Eq.24) states that visits of a vehicle in a route should not be overlapped, and there must be at least the traveling time of those visited customers between their visits. To provide the security of the vehicles, and differentiate the routes between periods, the model inserts at least a certain interval ϵ between the visits of the same customer in different periods using the constraint (Eq.25). The constraint (Eq.26) prohibits the waiting time for a vehicle by limiting its tour as the sum of the service time and travel time to the customers on the route.

4.3. Valid inequalities for the CP model

In order to strengthen the CP model presented in Section 4.3, the following two valid inequalities are added:

$$Next(vh_{k,p}, x_{i,k,p}, i, i) \neq j \quad \forall i, j \in C | i \neq j, b_i + s_i + t_{i,j} < a_j, k \in V, p \in P \tag{27}$$

$$Min\ Z = \sum_{p \in P} \sum_{k \in V} \sum_{i \in N} t_{i,Next(vh_{k,p}, x_{i,k,p}, i, i)}$$

4.4. Lower bound

To strengthen the aforementioned MILP and CP models, we also present a simple lower bound for the studied PVRP-TWTS as follows:

customers, while sets RC1 (8 instances) and RC2 (8 instances) include a combination of random and clustered settings. Instance sets of type 1 have a short scheduling horizon, allowing only a few customers to route. On the other hand, the instance sets of type 2 have a long scheduling horizon, permitting more customers to be served by

the same vehicle. Within a set, the instances vary with respect to the width of the time windows.

In order to use these benchmark sets, we include additional parameters to these instances as follows. We assume three periods for the planning horizon and $\epsilon =$
$\lfloor \epsilon_{up} * 0.5 \rfloor$ as proposed by [1]. The travel time between two nodes is equal to the corresponding Euclidean distance, which is rounded down to the nearest integer. Since the demand of each customer varies from period to period in our PVRP-TWTS, we generate the demand
$LB = \sum_{p \in P} (\sum_{j \in C}$

As stated in the aforementioned models, in each period
p, there must be arrival to each customer j from another node i except the node $n + 1$. Hence, taking the minimum traveling time for visiting customer j provides a lower bound for the traveling time to the customer j. Furthermore, in each period p, there must be at least one return to the depot $(n + 1)$ from one of the customers. Thus, taking the minimum traveling time for returning to the depot $n + 1$ provides a lower bound for the traveling time to the depot $n + 1$.

In the presented lower bound (LB) in (Eq.29), the sum of minimum traveling time of each customer and the depot over all the periods provides a lower bound for the total traveling time of the vehicles over all periods, i.e., the objective function of the PVRP-TWTS. In order to strengthen the proposed MILP and CP models, we also included this lower bound (LB) to the MILP and CP models by including a constraint $Z \geq LB$.

5. Experimental results

Experimental results are divided into two subsections. In the first part, the data generation procedure is explained in detail. Then, the computational results of the MILP and the CP models are compared in the second section.

5.1. Data generation

In order to evaluate the performance of the proposed MILP and CP models, we generate 280 instances by modifying the well-known benchmark set of Solomon
[3] for the VRP with time windows. Originally, there are 56 instances in this set [3], where there are six subsets for the instances. The given data for these instances is the number of customers, the vehicle capacity, Euclidean coordinates of the nodes (customers and the depot), demand and service time of uniform distribution in the range [1,50]. We also assume that there are two types of vehicles with different capacities: the first type of vehicle (large vehicle) has a capacity of 100 units, whereas the other (small vehicle) has a capacity of 75 units.

Originally, the benchmark set of Solomon [3] has 100 customers and 25 identical vehicles. Since exact solution approaches have been presented in this study, these instances are very large to test the performance of the mathematical models. Therefore, we cropped the original instances to have small-sized instance sets with varying customers from 5 to 15. Namely, we take the first n customers in the original instances, and we generate 56 instances with n customers, where $n =$
$\{5, 8, 10, 12, 15\}$. Note that there are 56 instances for each customer size, resulting in 280 instances in total. For each instance, we assume that the number of vehicles is equal to the number of customers. Note that there are $\lfloor n/2 \rfloor$ large vehicles in each instance, whereas the rest are small vehicles.

5.2. Computational results

The computational tests have been structured on five sets of instances, with client numbers ranging from five to fifteen, as stated in Section 5.1. The MILP and CP models are solved using CPLEX 12.8, which is programmed using the IBM ILOG CPLEX Optimization Studio. Each instance with a 30- or 60-minute time constraint on a Core i5, 3.20 GHz, 8 GB RAM machine achieves MILP and CP results. The one-hour time constraint allows only cases with five consumers to be optimally solved. For cases with a large number of customers, optimum results cannot be reached using the investigated PVRP-TWTS since it is NP-hard. Here is how each instance's relative percentage deviation (RPD) is calculated: the nodes, and the time frames of the nodes. Section 5.1 noted that for every customer size, there are six subsets of instances: R1, R2, RC1, RC2, C1, and C2. The number of clients, distribution of nodes, and scheduling horizon are used to group each subset of instances in the first column of the table. Take instance set R101-R112-5 as an example. It has 5 customers with a short scheduling horizon (type 1) and customers distributed randomly (R).

Table 1. The results for the MILP and CP models under 30 minutes time limit.

Instance Set - # *of Customers*	**MILP**		**CP**	
	# of Best	**ARPD**	**# of Best**	**ARPD**

R101-R112-*5*	12	0.00	10	0.46
R201-R211-*5*	11	0.00	10	0.24
RC101-RC108-*5*	8	0.00	7	0.17
RC201-RC208-*5*	8	0.00	8	0.00
C101-C109-*5*	9	0.00	8	0.09
C201-C208-*5*	8	0.00	8	0.00
R101-R112-*8*	8	0.93	7	1.36
R201-R211-*8*	6	1.98	7	1.13
RC101-RC108-*8*	5	1.84	3	1.99
RC201-RC208-*8*	6	1.59	3	2.44
C101-C109-*8*	6	0.28	8	0.23
C201-C208-*8*	5	1.36	8	0.00
R101-R112-*10*	6	1.61	8	2.05
R201-R211-*10*	4	3.49	8	1.00
RC101-RC108-*10*	3	4.54	5	1.25
RC201-RC208-*10*	2	3.86	6	0.31
C101-C109-*10*	3	7.23	8	0.37
C201-C208-*10*	3^{1}	4.32	8	0.00
R101-R112-*12*	1^{1}	18.19	11	0.08
R201-R211-*12*	0^{5}	19.30	11	0.00
RC101-RC108-*12*	0^{3}	8.76	8	0.00
RC201-RC208-*12*	1^{2}	11.81	7	0.34
C101-C109-*12*	1	24.35	8	0.39
C201-C208-*12*	1^{1}	19.17	8	0.00
R101-R112-*15*	2^{9}	7.73	10	1.06
R201-R211-*15*	0^{8}	14.78	11	0.00
RC101-RC108-*15*	1^{3}	38.11	7	0.44
RC201-RC208-*15*	0^{6}	30.00	8	0.00
C101-C109-*15*	1^{6}	19.21	8	0.49
C201-C208-*15*	0^{5}	32.10	8	0.00
Overall	**121**	**9.22**	**235**	**0.53**

The performances of the models are also analyzed under a longer time limit to see their changes on the solution quality. Table 2 summarizes the results of the MILP and CP models under 60 minutes time limit. As seen in Table 2, again, the CP model clearly outperforms the MILP model in terms of the overall ARPD value. Specifically, the overall ARPD value of the MILP is 9.18%, whereas the overall ARPD value of the CP model is 0.50%.

Moreover, the MILP model can find 130 out of the 280 best solutions, whereas the CP model can find 236 out of the 280 best solutions. Similar to the results under a
30 minutes time limit, the effectiveness of the CP model is clearly seen when the number of customers increases. Additionally, the MILP cannot obtain a feasible solution within the 60 minutes time limit for some of the instances with 12 and 15 customers, whereas the CP finds a feasible solution for all instances within the 60 minutes time limit.

Table 2. The results for the MILP and CP models under 60 minutes time limit.

Instance Set - # *of Customers*	MILP		CP	
	# of Best	ARPD	# of Best	ARPD
R101-R112-*5*	12	0.00	10	0.46
R201-R211-*5*	11	0.00	11	0.00
RC101-RC108-*5*	8	0.00	8	0.00
RC201-RC208-*5*	8	0.00	8	0.00
C101-C109-*5*	9	0.00	9	0.00
C201-C208-*5*	8	0.00	8	0.00
R101-R112-*8*	10	0.67	8	0.69
R201-R211-*8*	6	1.01	7	1.23
RC101-RC108-*8*	4	0.80	4	0.78
RC201-RC208-*8*	6	2.87	3	0.16
C101-C109-*8*	6	0.28	8	0.10
C201-C208-*8*	6	0.46	8	0.00
R101-R112-*10*	8	1.16	6	2.30
R201-R211-*10*	5	2.66	7	1.08
RC101-RC108-*10*	5	4.21	3	3.33
RC201-RC208-*10*	4	2.61	4	1.35
C101-C109-*10*	5	5.06	8	0.26
C201-C208-*10*	3	4.51	8	0.00
R101-R112-*12*	1^1	17.04	11	0.08
R201-R211-*12*	0^5	15.03	11	0.00
RC101-RC108-*12*	1	11.02	7	0.96
RC201-RC208-*12*	1^1	10.05	7	0.27
C101-C109-*12*	1	17.47	9	0.00
C201-C208-*12*	1^1	12.72	8	0.00
R101-R112-*15*	0^9	19.79	12	0.00
R201-R211-*15*	0^8	25.16	11	0.00
RC101-RC108-*15*	1^3	38.47	7	2.02
RC201-RC208-*15*	0^5	35.33	8	0.00
C101-C109-*15*	0^3	23.70	9	0.00
C201-C208-*15*	0^5	23.40	8	0.00
Overall	**130**	**9.18**	**236**	**0.50**

As seen in Table 2, ARPD values and the number of best solutions found do not have a dramatic change compared to the results under a 30 minutes time limit. Namely, the proportion of the best solutions found and the ARPD values of the MILP and CP models are nearly the same with those obtained under a 30 minutes time limit. However, according to the "# of Best" column's superscript numbers, as the solution time limit increases, the MILP model starts to find more feasible solutions; specifically, it finds nine more feasible results.

In terms of CPU times, the CP model has always operated up to the time limit, so the CPU time of the CP model is equal to the given time limit. This also applies to the MILP model, except for the instances with five customers. The MILP model has yielded results in less than the given time limit in some of the instances with five customers. Namely, the average CPU time for the instances with five customers is 524.67 seconds for the MILP model when the limit is 30 minutes, and it is
984.43 seconds when the limit is 60 minutes.

Table 3 presents the effect of the time limit on the number of best solutions found and the ARPD values. Instead of comparing the MILP and CP models under the same time limit, Table 3 compares the MILP and CP results under 30 minutes time limit with the best results obtained under the 60 minutes limit. As the time limit increases, the objective function decreases, and the solution quality increases, the best results are assumed as the best results obtained among MILP and CP models under 60 minutes time limit. Table 3 indicates the number of best solutions found and ARPD values of the MILP and CP models under 30 minutes time limit with respect to the best results. Note that these results and deviations are calculated with respect to the best solutions obtained under 60 minutes time limit.

Table 3. The results for the MILP and CP models under 30 minutes time limit with respect to the best results obtained under 60 minutes time limit.

Instance Set - # *of Customers*	MILP		CP	
	# of Best	ARPD	# of Best	ARPD
R101-R112-*5*	12	0.00	10	0.46
R201-R211-*5*	11	0.00	10	0.24
RC101-RC108-*5*	8	0.00	7	0.17
RC201-RC208-*5*	8	0.00	8	0.00
C101-C109-*5*	9	0.00	8	0.09
C201-C208-*5*	8	0.00	8	0.00
R101-R112-*8*	6	1.16	7	1.61
R201-R211-*8*	5	2.25	6	1.40
RC101-RC108-*8*	4	1.93	2	2.08
RC201-RC208-*8*	4	4.28	2	5.25
C101-C109-*8*	6	0.28	8	0.23
C201-C208-*8*	5	1.36	8	0.00
R101-R112-*10*	4	2.35	3	2.79
R201-R211-*10*	2	4.83	3	2.32
RC101-RC108-*10*	0	7.99	2	4.57
RC201-RC208-*10*	1	5.82	2	2.19
C101-C109-*10*	3	8.38	6	1.36
C201-C208-*10*	3^{1}	5.20	7	0.69
R101-R112-*12*	1^{1}	21.30	1	2.38
R201-R211-*12*	0^{5}	21.23	4	1.76
RC101-RC108-*12*	0^{3}	15.96	1	6.42
RC201-RC208-*12*	0^{2}	14.40	0	2.62
C101-C109-*12*	0	27.05	4	2.45
C201-C208-*12*	1^{1}	19.78	6	0.41
R101-R112-*15*	0^{9}	20.28	0	16.91
R201-R211-*15*	0^{8}	26.05	1	7.31
RC101-RC108-*15*	0^{3}	59.60	0	18.48
RC201-RC208-*15*	0^{6}	33.82	0	9.30
C101-C109-*15*	0^{6}	19.87	5	3.06
C201-C208-*15*	0^{5}	32.10	8	0.00
Overall	**101**	**11.91**	**137**	**3.22**

Table 3 indicates that the solution quality of both models gets worse when the time limit decreases to 30 minutes. The number of best solutions found by the MILP model decreases from 130 to 101, while its ARPD increases from 9.18% to 11.91%. The difference between the results is not too much for the MILP model because the CP model mostly finds the best solutions. In spite of obtaining the MILP results under different time limits, the performance of the MILP model is worse than the CP model in both configurations, resulting in close numbers in the solutions. Therefore, it seems that increasing the solution time limit does not change the performance of the MILP model significantly.

As seen from Table 3, when the time limit decreases, the number of best solutions provided by the CP model decreases from 236 to 137, and the ARPD increases from 0.50% to 3.22%. We can infer from those results that the level of a time limit has a vital effect on the performance of the CP model.

In order to compare all the results of both models under two different time limits, Figure 1 is depicted. In Figure 1, we compare the ARPD values of MILP and CP models under 30- and 60-minutes time limits with respect to the best results obtained among MILP and CP models under 60 minutes time limit. The detailed RPD values for all model and time limit configurations (MILP-30 min., MILP-60 min., CP-30 min., CP-60 min.) are also provided in the Appendix (Table A1-A5).

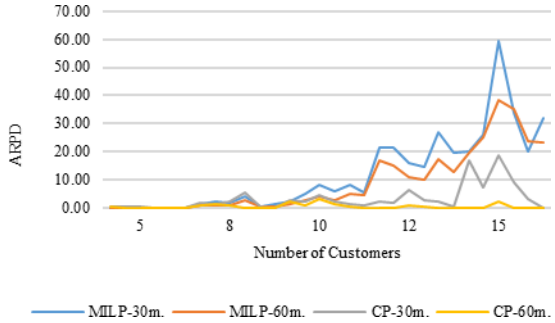


Figure 1. ARPD results of MILP and CP models under 30-and 60- minutes limits.

As seen from Figure 1, all ARPD results are nearly the same for the instances with 5 and 8 customers. The ARPD results start to be differentiated as the customer size increases, and the differences are obvious when the customer size equals 15. The CP model generates the best solutions under 60 minutes time limit, and the worst results belong to the MILP model under a 30 minutes time limit. The ARPD results of the CP model under 30 minutes time limit are much better than the ARPD results of the MILP model under 60 minutes time limit, highlighting the superior performance of the CP model over the MILP model.

As mentioned before, and it can also be seen from Figure 1, the level of a time limit has a vital effect on the performance of the CP model. To see the limitation of this effect, we increased the time limit to 120 minutes just for the CP model and obtained the results for some of the instance configurations. Note that the CP model dominates the MILP model for the instances with the larger sizes of the customers. Additionally, data sets having only five customers are not realistic when the real-life situation is considered. Therefore, we did not obtain the results under 120 minutes for the instances with five customers. Consequently, only a single representative instance from each combination is taken for the remaining data sets, resulting in 24 instances in total. For these instances, Figure 2 is generated to show the performance of the CP model under different time limits, i.e., 30, 60, and 120 minutes. In this figure, we compare the RPD values of the CP model under different time limits with respect to the best results obtained by the CP model under 120 minutes time limit.

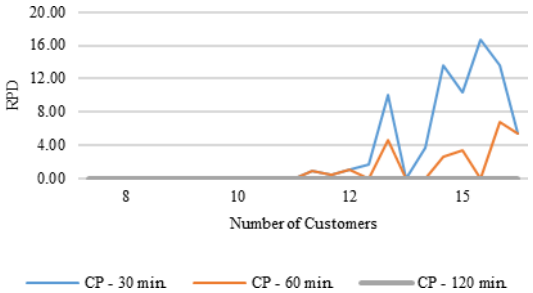


Figure 2. RPD results of the CP model under 30-, 60-, and 120- minutes limits.

As seen in Figure 2, for the instances with 8 and 10 customers, the CP model found the same result under all the time limitations. However, for the instances with 12 and 15 customers, Figure 2 indicates that as the time limit increases, the RPD value of the CP model decreases. The difference is not obvious for the smaller instances, but apparent when the customer size equals
15. There is a substantial difference between the solutions under the 30- and 60-minutes time limit. Nevertheless, there is no significant difference between the solutions under the 60- and 120-minutes time limit.

Finally, the detailed analysis of the experiments designates the superior performance of the CP model over the MILP model. As the given time limit increases, both models' solution quality improves, but more improvement is achieved for the CP model. However, there is not a noteworthy difference between the results under the 60- and 120-minutes time limit of the CP model. Consequently, it can be said that the CP model is very effective in solving the PVRP-TWTS in terms of the solution quality, as compared to the MILP model.

6. Conclusion

In order to distribute valuable products—in this case, money—among the city's ATMs in a secure manner, this study presents new MILP and CP models for the PVRP-TWTS. Waiting durations are not permitted in the PVRP-TWTS in order to prevent unprotected situations and ensure the protection of the transferred products. By imposing the time spread limits, the vehicle routes are differentiated across the planning periods. Furthermore, time frames take into account consumer availability, i.e., ATM operating and off hours. In light of these practical limitations,

the purpose of this article is to design routes that are both safe and efficient for the vehicles in question in terms of total trip time and fuel consumption. Within the realm of high-value shipment transportation, the suggested solution approaches can be utilized to build efficient and secure routes while taking into account various practical constraints like waiting time restrictions, time windows, and arrival time diversification. This is in regards to the managerial implications of the study. The manager may find efficient solutions to such complicated problems with the aid of the provided models. This research adds to the existing body of knowledge in the following ways. By including genuine inequalities into both mathematical models, we were able to enhance the solution quality and provide a lower limit for the issue under investigation. This problem's specifics informed the selection of cases from the literature, which were then adjusted. After then, the models' answers were evaluated by comparing them to the updated benchmark set. We can now find high-quality answers for modest numbers of clients up to 15, utilizing the provided precise solution techniques for such a complicated issue. This is a huge breakthrough, as shown by the results. To reiterate, the CP model is commercially available, uses a black-box solver, and is quite small. Despite the clear future benefits of the CP model, there is a risk of significant downsides, such as worse solution quality and increased CPU use due to an inevitable rise in client nodes.

References

[1] The authors of the cited article are Michellet, Prins, Amodeo, Yalaoui, and Vitry (2014). An iterative multi-start local search is used to the periodic vehicle routing issue, taking into consideration time frames and their limits on services. Publication: Computers & Operations Research,vol. 41, pages 196-207. Hoogeboom and Dullaert (2019) are cited as [2]. Vehicle routing with arrival time diversification. Volume 275, Issue 1, Pages 93–107, European Journal of Operational Research. (Solomon, 1987) [3]. Algorithms for the vehicle routing and scheduling issues with time window limitations. "Operations Research" (vol. 35, no. 2), pp. 254-265. The authors of this work are Rossi, F., Van Beek, P., and Walsh, T., from 2006. A guide on constraint programming. Published by Elsevier. [5] Shaw, P. (1998). Finding optimal routes for vehicles by combining constraint programming with local search techniques. Springer, Berlin, Heidelberg, 1998, 417-431. Michael Maher and Jean-Francois Puget, editors, Principles and Practice of Constraint Programming — CP98.

This information is sourced from Backer et al. (2000). Finding optimal routes for vehicles by combining constraint programming with metaheuristics. Volume 6, Issue 4, pages 501-523, Journal of Heuristics. This information is sourced from Guimarans et al. (2011). Finding optimal routes for vehicles by combining metaheuristics with constraint programming and Lagrangean relaxation. Volume 4, Issue 1, Pages 61–81, International Journal of Information Systems and Supply Chain Management. The authors of the cited study are Hojabri, Gendreau, Potvin, and Rousseau (2018). Rahman and Nielsen (2019) provided the cited information. Scheduling autonomous transport vehicles for material distribution systems. Publication: Applied Soft Computing, Volume 82, Issue 10, Page 10. In 2019, Emde and Zehtabian published a paper. Volume 57, Issue 5, pages 1315–1330, International Journal of Production Research.