The multi-trip vehicle routing problem with increasing profits for the blood transportation: An iterated local search metaheuristic

Andrea Pirabán-Ramírez, William Javier Guerrero-Rueda, Nacima Labadie

Version of Record: https://www.sciencedirect.com/science/article/pii/S0360835222003552
Manuscript_00f4563f693b56e91ff6a89d733b82e8

The multi-trip vehicle routing problem with increasing profits for the blood transportation: An iterated local search metaheuristic

Andrea Pirabán-Ramírez[a,b,c,*], William Javier Guerrero-Rueda[d], Nacima Labadie[b]

[a] *Doctorado en Logística y Gestión de Cadenas de Suministros, Universidad de La Sabana, Campus Puente del Común, Km. 7, Autopista Norte de Bogotá. Chía, Cundinamarca, Colombia.*
[b] *ICD-LIST3N, Université de Technologie de Troyes, 12 rue Marie Curie, CS 42060 10004 Troyes Cedex, France.*
[c] *Department of Productivity and Innovation, Universidad de la Costa, Calle 58 # 55 - 66, Barranquilla, Atlántico, Colombia*
[d] *Faculty of Engineering, Universidad de La Sabana, Campus Puente del Común, Km. 7, Autopista Norte de Bogotá. Chía, Cundinamarca, Colombia.*

Acknowledgments

This project was supported by the Fonds Européen de Développement Régional-FEDER (grant number OSTL-University of Technology of Troyes) and Universidad de La Sabana in Colombia (grant number INGPhD-9-2019).

*Corresponding author
Email address: andreapira@unisabana.edu.co (Andrea Pirabán-Ramírez)

The multi-trip vehicle routing problem with increasing profits for the blood transportation: An iterated local search metaheuristic

Abstract

This paper studies a multi-trip routing problem of a shuttle fleet to transport blood units from collection sites to a blood center. In this problem, the blood units intended to produce platelets and cryoprecipitate must be processed within eight hours from their donation and arrive at the blood center at a time less than its closing time to guarantee enough processing time. Since it is assumed that blood units are donated at a collection site following a constant ratio over its operating hours, this problem is modeled as a multi-trip vehicle routing problem with increasing profits for which a mixed-integer linear programming formulation is proposed. A hybrid iterated local search metaheuristic and an extended version are developed as solution methods. The extended version includes a mixed-integer linear programming component into the local search of the hybrid metaheuristic to optimize the decision on the departure times of the trips. The solution methods are tested on a new set of instances based on the blood collection system of Bogota, Colombia.

Keywords: Healthcare logistics, Blood supply chain, Perishable products, Variable profits, Hybrid metaheuristics, Optimization

1. Introduction

The blood, when extracted from the human body, is a perishable product that can be used for medical treatments such as surgery, organ transplantation, and cancer. The main blood products are whole blood (WB), red blood cells, platelets (PLTs), plasma, and cryoprecipitate (cryo) (American Red Cross, 2017). The last four products are known as the

Abbreviations: BC: blood center, BCP: blood collection problem, BKS: best-known solution, BSC: blood supply chain, cryo: cryoprecipitate, CS: collection site, ILS: iterative local search, MILP: mixed-integer linear programming, MT-VRPIP: multi-trip vehicle routing problem with increasing profits, PLT: platelet, TOP: team orienteering problem, VND: variable neighborhood descent, VRP: vehicle routing problem, WB: whole blood

main blood components and can be mechanically separated from a unit of WB. For details about methods to obtain the blood components, the reader is referred to the technical manual published by the American Association of Blood Banks (2014).

The *blood supply chain* (BSC) manages the flow of blood products from donors to patients through five echelons: donors, collection sites (CSs), blood centers (BCs), demand nodes or transfusion points, and patients (Pirabán et al., 2019). These echelons must be coordinated to perform the main processes: collection, transportation, testing, component processing, storage, and transfusion (Pirabán et al., 2019).

The BSC seeks to avoid two main problems: shortage and wastage. The shortage is unwanted as it may result in postponed surgeries, untreated patients, and deaths (WHO, 2017). Roberts et al. (2019) estimated with a statistical analysis that 119 of 195 (61%) countries did not have sufficient blood supply to satisfy their demand in 2017 because of limited donations. Additionally, discarded blood units generate a wastage cost considering both the effort expended in manufacturing and additional disposal processing (Custer et al., 2005). As reported by the WHO (2017), the average discard rate worldwide was 8.07% in 2013, especially because of the fact that 33% of units had passed their expiration date. Regarding the consequences of both shortage and wastage within the BSC, it is crucial to optimize the operational processes to increase the service level and reduce the wastage cost.

Transportation of WB units from CSs to BCs is performed by mobile CSs and shuttles. The mobile CSs are vehicles that collect blood units at frequented sites (e.g., parks, central stations, or universities) and then transport the blood units to a BC. Besides, shuttles are used to transport the collected units from fixed CSs or support the transportation activities of mobile CSs (Sahinyazan et al., 2015). This transportation process should be done on the same day of collection to ensure proper refrigeration of WB units.

The transportation process must meet two main constraints. First, the WB units to obtain PLTs and cryo must be processed within 8 h from their donation time (American Association of Blood Banks, 2014). This time limit is called the *processing time limit*. Second, the WB units intended to produce PLTs and cryo must be delivered to the BC at a time considerably less than its closing time, called the *arrival time limit*, to guarantee that the BC would have enough time to test and process the WB units.

Therefore, this paper seeks to answer the following question: *How to optimize the routing of shuttles transporting WB units from CSs to a BC and considering the processing and arrival time limits?*. The planning of shuttle routes, including the processing and arrival time limits, has not been addressed in the BSC literature to the best of our knowledge. Also, the risk of expiration is reduced since the time between the collection and processing is controlled when considering these time constraints. Hence, the general objective of this paper is to develop a mixed-integer linear programming (MILP) formulation and a metaheuristic to optimize the routing of shuttles considering the processing and arrival time limits.

The problem studied in this paper assumes that the CSs are located in a set of scattered nodes. Each CS has a WB donation level or profit, which is collected at a constant ratio over the operating hours of the CS. All WB units collected by the CSs must be delivered to the BC using a fleet of shuttles. For this activity, each shuttle can perform up to two consecutive trips. The first trips visit the necessary CSs to supply the demand for WB units intended to produce PLTs and cryo. Hence, the first trips must respect the processing and arrival time limits. If the number of WB units collected with the first trips is less than the demand, a shortage cost is imposed per missing unit. The second trips of the shuttle fleet are used to pick up the remaining units in each CS. This problem is denoted as the multi-trip vehicle routing problem with increasing profits (MT-VRPIP), which has the objective of minimizing the shortage, transportation, and delay costs.

The MT-VRPIP is a variant of the well-known vehicle routing problem (VRP), which was introduced by Dantzig & Ramser (1959) and is extensively studied by several authors. General surveys on the VRP and its variants can be found in Golden et al. (2008), Eksioglu et al. (2009), and Vidal et al. (2020). Additionally, the MT-VRPIP is NP-hard since it is an extension of the VRP, which is well-known to be NP-hard. It means no exact methods are providing optimal solutions in polynomial time for any size of the problem.

Since metaheuristics provide high-quality solutions on larger problems (Toth & Vigo, 2014), a hybrid iterated local search (hybrid-ILS) framework is developed as a solution method for the MT-VRPIP. This method determines the departure time of the shuttle trips within the local search component based on a set of possible times. It is required since the departure time of each shuttle trip can significantly change the number of collected donations

given the increasing profits. Additionally, one extension of the hybrid-ILS framework is proposed. In this extension, the local search component includes a MILP model to optimize the subset of decisions modeling the departure times of shuttle trips with fixed routing variables. This extension is named the hybrid-ILS+MILP. The developed solution methods are tested on a new set of instances based on the blood collection system of Bogota, Colombia.

The paper has several contributions. First, it provides a formalization of the MT-VRPIP, which includes WB units to produce all the blood components, processing and arrival time limits, increasing linear functions to represent donations at CSs, and multiple trips for the shuttle fleet. Second, it presents two metaheuristic frameworks based on the ILS to solve the MT-VRPIP. Third, it describes a new set of instances for the MT-VRPIP based on the blood collection system of Bogota, Colombia. Fourth, the experimental results given in this paper demonstrate the applicability of the methods and provide the best-known solutions (BKSs), which can be a point of comparison to other studies.

The rest of the document is structured as follows. In Section 2, a review of related literature is discussed. In Section 3, the MT-VRPIP and its corresponding mathematical model are presented. The hybrid-ILS and hybrid-ILS+MILP metaheuristics proposed as solution methods are described in Section 4. The testing instances and computational results are shown in Section 5. Finally, Section 6 provides conclusions and future research directions.

2. Literature review

Works related to the problem studied in this paper are reviewed under two groups: (i) VRPs focusing on the transportation of blood units from CSs to BCs also known as the *blood collection problem* (BCP) and (ii) other related vehicle routing variants. The literature related to the first group is searched using the scope and review methodology proposed by the survey of Pirabán et al. (2019), but considering scientific research published between 2005 and 2021. Interested readers are also referred to Beliën & Forcé (2012) and Osorio et al. (2015) for surveys on BSC management.

2.1. Transportation of donated blood units from CSs to BCs

The BCP was introduced by Prastacos (1984). The BCP selects the CSs to be visited by a vehicle fleet to achieve a collection target and designs the routes of the fleet to minimize the transportation cost. Variants of the BCP are reviewed under two subgroups: (i) BCPs that design vehicle routes to visit fixed CSs in a single period and (ii) BCPs that locate CSs at potential sites and design vehicle routes to visit the located CSs over a planning horizon.

Within the first subgroup, four articles are found in the literature (Doerner et al., 2008; Ghandforoush & Sen, 2010; Mobasher et al., 2015; Özener & Ekici, 2018). The four papers studied the routing problem of an uncapacitated fleet of shuttles to transfer blood units to a BC. They considered multiple visits to each CS in the same period and included the processing time limit. These authors assumed uncapacitated vehicles since blood bags are small compared to the capacity of shuttles. Additionally, some of these papers allowed a single trip to each shuttle (Doerner et al., 2008; Ghandforoush & Sen, 2010) and others allowed multiple trips to each shuttle (Mobasher et al., 2015; Özener & Ekici, 2018).

Doerner et al. (2008) assumed that a shuttle can pick up all the blood units collected in a period by a CS when the shuttle visits the CS within a fixed time window at the end of the period. Contrary, Ghandforoush & Sen (2010) stated a number of blood units that can be collected from a CS when a shuttle visits it at fixed times during the period and attempted meeting demand with all blood units collected by the shuttles. The objectives were to minimize the traveled distance for Doerner et al. (2008) and the production and transportation costs for Ghandforoush & Sen (2010). The former developed a MILP model, while the latter proposed a branch-and-bound algorithm and several constructive heuristics.

The *integrated collection and appointment-scheduling problem* introduced by Mobasher et al. (2015) and the *maximum blood collection problem* exposed by Özener & Ekici (2018) allowed multiple trips per shuttle. Both papers proposed a cluster-first-route-second heuristic as a solution method. Mobasher et al. (2015) focused on scheduled donors, which allow the decision-makers to plan staff and resources effectively. In contrast, Özener & Ekici (2018) focused on walk-in donors, which are the most common in organizations, like the MT-VRPIP.

The MT-VRPIP mainly differs from the problems exposed by Mobasher et al. (2015) and Özener & Ekici (2018) in five aspects. First, the MT-VRPIP involves transporting

WB to produce all the main components, while the other problems focused uniquely on PLTs. Consequently, the MT-VRPIP can force compliance with the processing time limit only to extract PLTs and cryo and relax this constraint for the other products. Second, the MT-VRPIP imposes a minimal quantity of WB units, which must meet the processing time limit. In contrast, the other problems sought to maximize the number of WB units collected by shuttles without considering demand. Third, the MT-VRPIP guarantees that the trips intended to meet the processing time limit reach a BC before the arrival time limit in contrast to the other problems. Therefore, the MT-VRPIP ensures the BC will have enough time to process the WB units before the end of the period. Fourth, Mobasher et al. (2015) and Özener & Ekici (2018) used the scheduled appointments of donors and an irregular pattern of donations, respectively, to calculate the number of WB units collected by a shuttle when it visits a CS. The irregular pattern consists of a different donation level per time slot within the operating hours of each CS. Contrary, the MT-VRPIP calculates the number of WB units collected by a shuttle based on a constant ratio of donations per hour over the operating hours of each CS. Finally, MT-VRPIP avoids the constraint proposed by Mobasher et al. (2015) and Özener & Ekici (2018) that assigns the same shuttle to visit a fixed cluster of CSs. According to these authors, this assumption makes the solution more practical since a driver visits the same set of locations and recognizes the area. However, since in the MT-VRPIP, the trips do not necessarily visit all the CSs, this assumption is unpractical. Moreover, eliminating this constraint could lead to solutions that minimize transportation costs but increasing the problem complexity.

Within the BCPs variants that locate CSs over a planning horizon, one paper without shuttles (Gunpinar & Centeno, 2016) and two papers using shuttles to pick up the collected blood (Sahinyazan et al., 2015; Rabbani et al., 2017) are found. In Gunpinar & Centeno (2016), a mobile CS could visit several locations each period to collect their uncertain supply and meet demand. The different locations of the mobile CS define its routing. A robust optimization approach and a branch-and-price algorithm were presented to solve the problem. Sahinyazan et al. (2015) designed the routes of a limited-uncapacitated fleet of shuttles assuming that a CS can receive a single visit of a shuttle at the end of the collection day. They decided the number of periods each CS is located at a potential site and proposed a

two-stage heuristic to solve the problem. Rabbani et al. (2017) assumed that a shuttle, from a capacitated-homogeneous fleet, must arrive at a CS within a fixed time window. First, they presented a fuzzy mathematical programming model, which locates the CSs considering a fuzzy PLT potential of possible locations. Second, they implemented a simulated annealing algorithm to design the routes of shuttles. Since the objective for Gunpinar & Centeno (2016) and Sahinyazan et al. (2015) was to minimize the traveled distance, they included a constraint to ensure demand fulfillment. In contrast, since Rabbani et al. (2017) excluded demand, they maximized the collected blood units and minimized the operational costs.

To the best of our knowledge, no articles in the literature jointly consider WB to produce all blood components, an increasing linear function to represent donations at CSs, the processing and arrival time limits, and a shuttle fleet that can perform multiple trips. Additionally, no article has developed a matheuristic (hybrid-ILS+MILP) as a solution method. Therefore, the MT-VRPIP addressed in this article, its mathematical formulation, and the proposed solution methods represent a contribution to the BSC literature.

2.2. Related vehicle routing problems

The MT-VRPIP is related to two variants of the VRP: the *multi-trip VRP* and the *team orienteering problem* (TOP). The multi-trip VRP, first exposed by Fleischmann (1990), allows multiple trips to each vehicle during a period. Cattaruzza et al. (2016) presented a review of the multi-trip VRP, which includes mathematical formulations, solution methods, and variants. The multi-trip VRP variant more related to the MT-VRPIP is the multi-trip VRP with time windows in which the vehicle fleet should visit each customer within a time interval (Hernandez et al., 2016; François et al., 2019; Neira et al., 2020; Pan et al., 2021).

In the TOP, visiting all clients is not mandatory. Therefore, a profit is associated with each customer that makes such a customer more or less attractive (Chao et al., 1996). This problem first decides the customers to visit based on profits; then, it decides the vehicle routes to serve the selected customers (Toth & Vigo, 2014; Tsakirakis et al., 2019; Hammami et al., 2020; Panadero et al., 2020; Xu et al., 2021). The reader is also referred to the survey on the orienteering problem of Gunawan et al. (2016).

The main difference between the MT-VRPIP and the VRP variants mentioned up to

now is the effect of removing a node or changing a departure time. When a node is removed in the multi-trip VRP, a noncompliance in the demand of the node can be generated. In contrast, if it is removed in the TOP and in the MT-VRPIP, the total profits collected by the vehicle fleet can change. In the case of the TOP, the profit of a node is counted if the node is visited. In the MT-VRPIP, the magnitude of the collected WB units also depends on the time the node is visited by a shuttle. Additionally, if the departure time of a trip changes in the VRP variants with time windows, it can affect the fulfillment of the time windows. Conversely, changing the departure time of the first trip of a shuttle in the MT-VRPIP can change the collected blood units because of the increasing profits.

Furthermore, the TOP with variable profits are classified into two groups: (i) profits that depended on the arrival time at each vertex (Murat Afsar & Labadie, 2013) and (ii) profits that depended on the service time at each vertex (Yu et al., 2019; Kim et al., 2020). The MT-VRPIP is related to the first group because of the increasing profits. In contrast, Murat Afsar & Labadie (2013) associated the profit to a decreasing function of time.

To the best of our knowledge, the VRP variants mentioned in this section excluded either the processing time limit or the increasing profit of a node over its time window. Therefore, the MT-VRPIP represents a contribution to the literature on the mentioned VRP variants.

3. Problem statement and model formulation

The MT-VRPIP is defined on a weighted and directed graph $G = (V', A)$ with a set of vertices $V' = \{V \cup \{0, n+1\}\}$ and a set of arcs $A = \{(i, j) | i, j \in V', i \neq j\}$. V is the set of n CSs, and the nodes $\{0, n+1\}$ denote the BC or depot as the starting and ending points of the routes to build, respectively. Each node $i \in V'$ is characterized by a non-negative profit p_i, a service time s_i, and a time interval of activity $[e_i, l_i]$ being e_i the opening time and l_i the closing time. The profit represents the number of WB units that may be collected at the node during its time interval of activity. By definition, the BC profits p_0 and p_{n+1} equal zero. As shown in Fig. 1, the WB units collected at node i increases in the interval (e_i, l_i) with the rate $\lambda_i = p_i/(l_i - e_i)$. Thus, λ_i equals zero in the interval $[0, e_i]$ and p_i in the interval $[l_i, +\infty)$. Finally, $(i, j) \in A$ represents the arc linking nodes i and j with a travel time c_{ij}.

A set F of u uncapacitated shuttles located at the BC is used to pick up the WB units

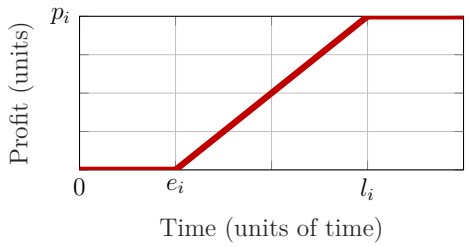

Figure 1: Profit of node i as a function of time

from the CSs and deliver them to the BC. Each shuttle $f \in F$ can perform two types of trips grouped in the set $K = \{\text{st}, \text{nd}\}$. Hereafter, *st* and *first-type trips* will refer to one first-type trip and the first-type trips of the u shuttles, respectively. Likewise, *nd* and *second-type trips* will refer to one second-type trip and the second-type trips of the u shuttles, respectively.

The first-type trip of shuttle $f \in F$ starts at the BC at time $t^{\text{st}}_{0f} \geq e_0$ and ends at the BC at time $t^{\text{st}}_{n+1,f}$. The service starting time t^{st}_{jf} at node $j \in V \cup \{n+1\}$ is at least the service starting time of the previous node $i \in V \cup \{0\}$ in the trip f plus the service time e_i of node i plus the travel time c_{ij} between nodes i and j. Each CS is visited at most once by the first-type trips. If CS $i \in V$ is visited by the first-type trip of shuttle f at time $t^{\text{st}}_{if} \geq e_i$, the shuttle picks up a quantity of WB units $y^{\text{st}}_{if} = \min\{\lfloor \lambda_i(t^{\text{st}}_{if} - e_i) \rfloor, p_i\}$. The CSs visited by the first-type trips constitute the set $V^{\text{st}} \subset V$ and provide the quantity $y^{\text{st}}_{\text{total}} = \sum_{i \in V^{\text{st}}} \sum_{f=1}^{u} y^{\text{st}}_{if}$. The first-type trips supply a demand q with their collected WB units $y^{\text{st}}_{\text{total}}$.

This demand has two requirements. First, the demand must be supplied by WB units with a maximum age $a^{\max}$, which represents the processing time limit. Therefore, the age a_i of WB units collected at CS $i \in V$ by shuttle $f \in F$ when they arrive at the BC at time $t^{\text{st}}_{n+1,f}$ must be less than or equal to $a^{\max}$. It is assumed that the age a_i starts to count at the beginning of activities at CS i, i.e., $a_i = t^{\text{st}}_{n+1,f} - e_i \leq a^{\max}$. Second, the demand must be delivered to the BC before the arrival time limit $t^{\max}$ with $e_{n+1} \leq t^{\max} \leq l_{n+1}$. This constraint states that the first-type trip of shuttle $f \in F$ must arrive at the BC at a time less than or equal to $t^{\max}$, i.e., $t^{\text{st}}_{n+1,f} \leq t^{\max}$.

The second-type trips pick up the remaining profit from CSs. The second-type trip of shuttle $f \in F$ starts at the BC at time $t^{\text{nd}}_{0,f}$ greater than the end of its first-type trip, and ends at the BC at time $t^{\text{nd}}_{n+1,f}$. Each CS is visited at most once by the second-type trips. CS $i \in V$ is visited by a second-type trip at a time t^{nd}_{if} greater than or equal to its closing

time l_i if there is a remaining profit to collect, i.e., if $\sum_{f=1}^{u} y_{if}^{\text{st}} < p_i$. If CS i is visited by a second-type trip, the shuttle f picks up a quantity of WB units $y_{if}^{\text{nd}} = p_i - \sum_{f=1}^{u} y_{if}^{\text{st}}$.

According to the description up to this point, each CS must receive a visit from a first- or second-type trip at a time greater than or equal to its closing time of activity to collect its total profit. Also, the second-type trips must deliver the WB units to the BC regardless of the BC closing time. Therefore, the MT-VRPIP accepts delays at both the CSs and BC. Allowing these delays makes the MT-VRPIP a problem with soft time windows (see details on soft time windows in Xia & Fu (2019) and He et al. (2021)). Note that no delay to the BC for the first-type trips is possible due to the arrival time limit.

Although all delays are calculated in the MT-VRPIP, only two delay types are penalized. First, the maximum delay of each first- and second-type trip considering only the delays on the CSs. Second, the BC delay by the second-type trip of shuttle f denoted as $\beta_{n+1,f}^{\text{nd}} = \max\{t_{n+1,f}^{\text{nd}} - l_{n+1}, 0\}$. Only the maximum delay on CSs is penalized to minimize the maximum time that the staff at CSs must wait once the collection is finalized to deliver the WB units to the shuttle. Likewise, the BC delay is penalized to minimize the maximum time that the BC staff must wait to receive the WB units from the second-type trips.

The MT-VRPIP considers a transportation cost ω^t per unit of time, shortage cost ω^s per WB unit, and delay cost ω^d per unit of time to penalize the travel time, shortage, and delays, respectively. The objective of the MT-VRPIP is to minimize the total cost determining (i) at most u first-type trips, which visit at most once each CS, to supply the demand q of WB units intended to produce PLTs and cryo and (ii) at most u second-type trips to pick up the remaining profits subject to time limitations. The routing decisions are noted by boolean variables x_{ijf}^{k} taking the value 1 if arc (i, j) is traversed by trip type k of shuttle f. The service starting time at node i by trip type k of shuttle f is denoted by real variables t_{if}^{k}. The routing and visit time decisions set the collected WB units and the delay at node i by trip type k of shuttle f, which are denoted by integer variables y_{if}^{k} and real variables β_{if}^{k}, respectively. Additionally, the units of unsatisfied demand, noted by real variable d, and the maximum delay of shuttle f in trip type k considering delays on CSs, denoted by real variables δ_{f}^{k}, are set. For easy reference, the notations of this paper are listed in Table 1.

For a proper understanding of the problem, an example is provided in Fig. 2. The

Sets	
V	Set of CSs, $V = \{1, ..., n\}$
V'	Set of CSs and BC nodes, $V' = \{V \cup \{0, n+1\}\}$
A	Set of arcs, $A = \{(i,j) \,\vert\, i, j \in V', i \neq j\}$
F	Set of shuttles, $F = \{1, ..., u\}$
K	Set of trip types that each shuttle may perform, $K = \{\text{st}, \text{nd}\}$
Parameters	
n	Number of CSs
u	Number of shuttles
M	Big value
p_i	Total donation level or profit of node i [WB units]
s_i	Service time at node i [units of time]
e_i	Opening time of activities at node i [units of time]
l_i	Closing time of activities at node i [units of time]
λ_i	Collection rate of node i, $\lambda_i = p_i/(l_i - e_i)$ [WB units/units of time]
c_{ij}	Travel time between nodes i and j [units of time]
q	Demand [WB units]
$t^{\max}$	Arrival time limit at the BC [units of time]
$a^{\max}$	Processing time limit [units of time]
ω^{t}	Traveling cost per unit of time
ω^{s}	Shortage cost per WB unit
ω^{d}	Delay cost per unit of time
Decision variables	
x^k_{ijf}	1 if arc (i,j) is traversed by shuttle f in trip type k, 0 otherwise
t^k_{if}	Service starting time at node i by shuttle f in trip type k
Auxiliary variables	
y^k_{if}	Collected WB units at node i by shuttle f in trip type k
d	Units of unsatisfied demand q
β^k_{if}	Delay at node i by shuttle f in trip type k
δ^k_f	Maximum delay of shuttle f in trip type k on CSs nodes

Table 1: Parameters and variables of the MT-VRPIP model.

instance presented in the example is one of the instances proposed for the MT-VRPIP (see Section 5.1 for details). The solution of the example is optimal and was obtained with the software CPLEX V 12.8. The example presents a set of seven nodes comprising five CSs $\{1, ..., 5\}$ and two BCs $\{0, 6\}$, which are merged into the node 0 in Fig. 2. In Fig. 2 (top), each node $i \in \{0, ..., 6\}$ is presented with its total profit p_i, service time s_i, and time interval of activity $[e_i, l_i]$. For the node 4, its collection rate λ_4 equals 0.025 WB units per min since $\lambda_4 = p_4/(l_4 - e_4) = 9/(840 - 480) = 0.025$. If a shuttle visited the node 4 at the minutes 480 or 840, it would pick up 0 or 9 WB units, respectively. Additionally, the instance of the example assumes having two shuttles, a demand q of 54 WB units, a processing time limit

$a^{\max}$ of 480 min or 8 h, and an arrival time limit $t^{\max}$ equals to 772 min.

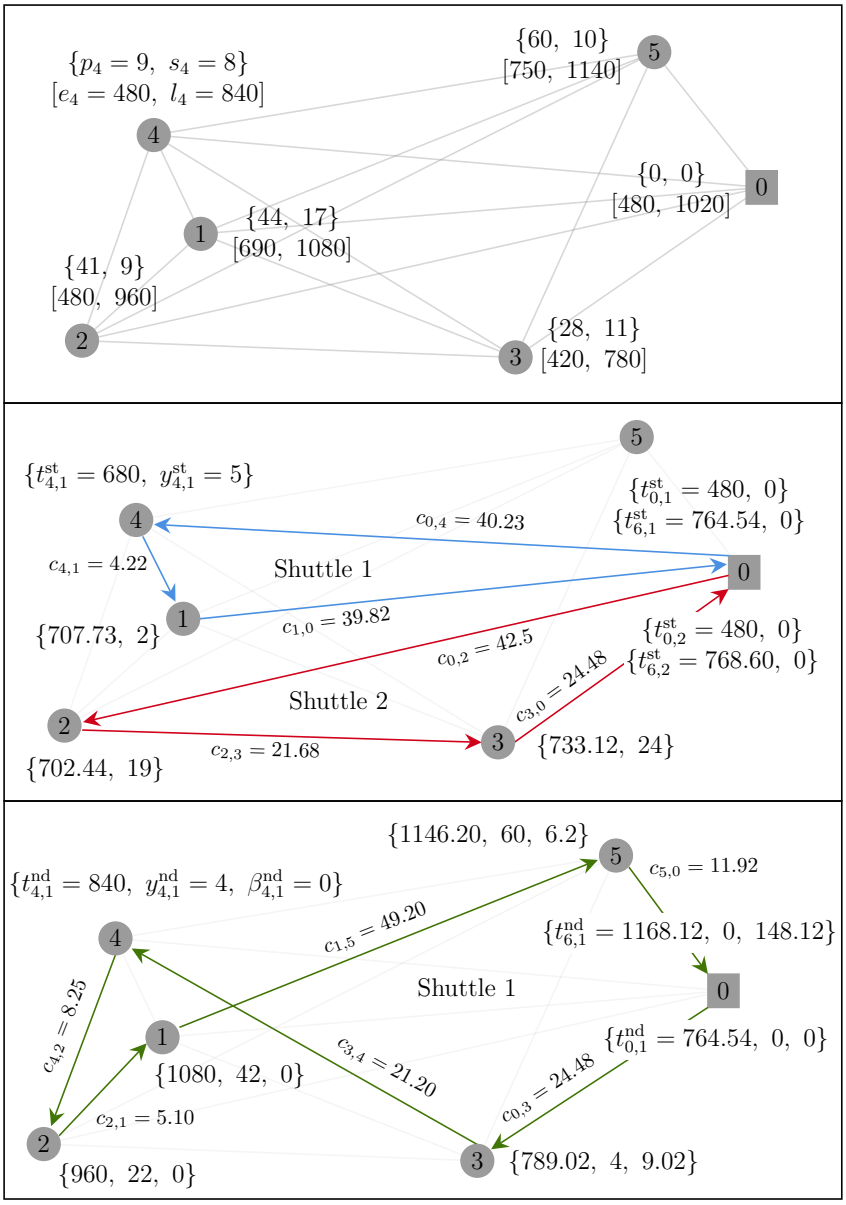

Figure 2: Example for the problem statement. **(top)** Parameters. **(middle)** Solution for the first-type trips. **(bottom)** Solution for the second-type trips. The notation is equivalent to that presented in Table 1. Node 6 is a copy of node 0. The units of time are minutes.

Fig. 2 (middle) presents the first-type trips. For instance, the shuttle 1 follows the trip $\pi_1^{\text{st}} = \langle 0, 4, 1, 6 \rangle$ starting and ending at the BC at the minutes $t_{0,1}^{\text{st}} = 480$ and $t_{6,1}^{\text{st}} = 764.54$, respectively, with $t_{0,1}^{\text{st}} \geq e_0$. The service starts at the node 4 at the minute $t_{4,1}^{\text{st}} = 680$ with $t_{4,1}^{\text{st}} \geq e_4$. Therefore, the shuttle 1 picks up at the node 4 the quantity $y_{4,1}^{\text{st}}$ equals to 5 WB units since $y_{4,1}^{\text{st}} = \min\{\lambda_4(t_{4,1}^{\text{st}} - e_4), p_4\} = \min\{0.025(680 - 480), 9\} = 5$. Note that

although the arrival time at the node 4 would be at the minute 520.23, since $t^{\text{st}}_{0,1} + s_0 + c_{0,4} = 480 + 0 + 40.23 = 520.23$, the shuttle waits 159.77 min at the node 4 to start the service at the minute 680 and collect more WB units. This happens also at the nodes 1 and 2, while at the node 3 the shuttle does not wait. The solution presented in Fig. 2 generates a shortage equals to 4 WB units since the quantity collected by the first-type trips $y^{\text{st}}_{\text{total}}$ is 50 WB units.

The shortage could not be less, mainly due to time constraints. For example, the node 5 is excluded from first-type trips since picking up at least one WB unit of its profit represents a failure to meet the arrival time limit $t^{\max}$ of 772 min. Specifically, a shuttle $f \in F$ would pick up one WB unit at the node 5 if it started the service at minute 756.5. This time is calculated according to the collection rate of the node. Therefore, shuttle f would return to the BC at minute 778.42 since $t^{\text{st}}_{5,f} + s_5 + c_{5,0} = 756.5 + 10 + 11.92 = 778.42$. Also, the arrival time limit $t^{\max}$ may prevent shuttles from picking up more WB units at the visited nodes. For example, to pick up 3 units instead of 2 units at the node 1, the service at this node should start at minute 716.59 instead of at minute 707.73. However, starting the service at the minute 716.59 would cause the shuttle 1 to arrive at the BC in a longer time than $t^{\max}$. Therefore, the shuttle 1 arrives at the BC a few minutes earlier than $t^{\max}$ because it is infeasible to pick up an additional unit at the node 1.

The shuttles meet the arrival time limit of 772 min since they finish their first-type trips at the minutes 764.54 and 768.60. Besides, the age of the WB units collected at the node 4 when they reach the BC equals 284.54 min since $a_4 = t^{\text{st}}_{6,1} - e_4 = 764.54 - 480 = 284.54$; then, the processing time limit is met. The reader can verify that the age of the WB units collected from the other nodes visited by the first-type trips meets the processing time limit.

In Fig. 2 (bottom), one second-type trip $\pi^{\text{nd}}_1 = \langle 0, 3, 4, 2, 1, 5, 6 \rangle$ is proposed, which starts after the end of its first-type trip at the minute $t^{\text{nd}}_{0,1} = 764.54$ and ends at the BC at the minute $t^{\text{nd}}_{6,1} = 1168.12$. The start of the service at CSs $\{1, ..., 5\}$ is given in a time greater than or equal to their closing times. All nodes must be visited by the second-type trip since the WB units collected at each node by the first-type trips is less than its total profit.

In the first-type trips presented in Fig. 2 (middle), all the nodes are visited within their time windows so no delays are generated. Additionally, along route π^{nd}_1, there is a delay at the node 3 of 9.02 min and node 5 of 6.20 min. However, only the maximum delay between

these two delays is counted. Since the route π_1^{nd} reaches the BC at the minute $t_{6,1}^{\mathrm{nd}} = 1168.12$ when its closing time is the minute $l_6 = 1020$, the BC delay is $\beta_{6,1}^{\mathrm{nd}} = 148.12$ min.

The travel time of the first-type trips equals 172.92 min since $(40.23 + 4.22 + 39.82 + 42.5 + 21.68 + 24.48) = 172.93$ and the travel time of the second-type trips equals 120.15 min since $(24.48 + 21.20 + 8.25 + 5.10 + 49.20 + 11.92) = 120.15$. In summary, the total travel time of the first- and second-type trips, shortage, maximum delay, and delay at the BC equal 293.07 min, 4 WB units, 9.02 min, and 148.12 min, respectively. Finally, the total cost of the example equals 5036.80 USD assuming transportation, shortage, and delay costs equal to 5 USD/min, 500 USD/WB unit, and 10 USD/min, respectively.

The following mixed-integer nonlinear programming model is stated for the MT-VRPIP,

$$\text{Min} \quad \omega^{\mathrm{t}} \sum_{(i,j)\in A} \sum_{f\in F} \sum_{k\in K} c_{ij} x_{ijf}^{k} + \omega^{\mathrm{d}} \sum_{f\in F} \sum_{k\in K} \left(\beta_{(n+1)f}^{k} + \delta_{f}^{k}\right) + \omega^{\mathrm{s}} d \tag{1}$$

$$\sum_{j\in V} x_{0jf}^{k} = \sum_{i\in V} x_{i(n+1)f}^{k} \leq 1 \quad f \in F; k \in K \tag{2}$$

$$\sum_{i=0}^{n} x_{ijf}^{k} = \sum_{i=1}^{n+1} x_{jif}^{k} \quad j \in V; f \in F; k \in K \tag{3}$$

$$\sum_{i=0}^{n} \sum_{f=1}^{u} x_{ijf}^{k} \leq 1 \quad j \in V; k \in K \tag{4}$$

$$t_{if}^{k} + s_i + c_{ij} - M\left(1 - x_{ijf}^{k}\right) \leq t_{jf}^{k} \quad (i,j) \in A; f \in F; k \in K \tag{5}$$

$$t_{if}^{\mathrm{st}} \geq e_i \sum_{j=1}^{n+1} x_{ijf}^{\mathrm{st}} \quad i \in V' \backslash \{n+1\}; f \in F \tag{6}$$

$$t_{(n+1)f}^{\mathrm{st}} \leq t^{\max} + M - M \sum_{i=1}^{n} x_{i(n+1)f}^{\mathrm{st}} \quad f \in F \tag{7}$$

$$t_{(n+1)f}^{\mathrm{st}} - e_i - M\left(2 - \sum_{j=1}^{n+1} x_{ijf}^{\mathrm{st}} - \sum_{j=1}^{n} x_{j(n+1)f}^{\mathrm{st}}\right) \leq a^{\max} \quad i \in V; f \in F \tag{8}$$

$$t_{if}^{\mathrm{nd}} \geq l_i \sum_{j=1}^{n+1} x_{ijf}^{\mathrm{nd}} \quad i \in V; f \in F \tag{9}$$

$$t_{0f}^{\mathrm{nd}} \geq t_{(n+1)f}^{\mathrm{st}} + s_{n+1} - M + M \sum_{i\in V} x_{i(n+1)f}^{\mathrm{st}} \quad f \in F \tag{10}$$

$$t_{0f}^{\mathrm{nd}} \geq e_0 \sum_{j \in V} x_{0jf}^{\mathrm{nd}} \quad f \in F \tag{11}$$

$$t_{if}^{k} - \beta_{if}^{k} \leq l_i + M - M \sum_{j=0}^{n} x_{jif}^{k} \quad i \in V' \backslash \{0\}; f \in F; k \in K \tag{12}$$

$$\delta_{f}^{k} \geq \beta_{if}^{k} \quad i \in V; f \in F; k \in K \tag{13}$$

$$y_{if}^{\mathrm{st}} = \lfloor \lambda_i \left(t_{if}^{\mathrm{st}} - e_i \right) \rfloor \quad i \in V; f \in F \tag{14}$$

$$y_{if}^{k} \leq p_i \sum_{j=0}^{n} x_{jif}^{k} \quad i \in V; f \in F; k \in K \tag{15}$$

$$\sum_{f=1}^{u} \sum_{k \in K} y_{if}^{k} = p_i \quad i \in V \tag{16}$$

$$\sum_{i \in V} \sum_{f=1}^{u} y_{if}^{\mathrm{st}} \geq q - d \tag{17}$$

$$x_{ijf}^{k} \in \{0, 1\} \quad (i, j) \in A; f \in F; k \in K \tag{18}$$

$$t_{if}^{k}, \beta_{if}^{k}, \delta_{f}^{k}, d \in \mathbb{R}^{+} \quad i \in V'; f \in F; k \in K \tag{19}$$

$$y_{if}^{k} \in \mathbb{Z}^{+} \quad i \in V'; f \in F; k \in K \tag{20}$$

The objective function (1) minimizes the total cost comprising the transportation, delay, and shortage costs. Constraints (2) guarantee that if a shuttle leaves the vertex 0 it must end at node $n+1$. Constraints (3) set the flow conservation of each trip; i.e., if a node j is visited, it must have a precedent node and a successor node. In addition, each CS is visited at most once by each type of trip because of the constraints (4). Constraints (5) define the visiting time at each node. The respect of time limitation on the first-type trips is guaranteed by the constraints (6)-(8). Constraints (9)-(11) guarantee the time limitation on the second-type trips. Constraints (12) calculate the delays and constraints (13) set the maximum delay δ_f^k of each trip considering the delays on CSs. The quantity of WB units collected by shuttles in the first- and second-type trips is calculated using constraints (14)-(16). Constraints (17) are related to demand satisfaction. Finally, constraints (18)-(20) fix the nature of variables.

To avoid the floor function in Constraints (14), the following constraints are proposed

instead,

$$y_{if}^{\text{st}} \leq \lambda_i \left(t_{if}^{\text{st}} - e_i \right) \quad i \in V; f \in F \tag{21}$$

Constraints (21) establish that the quantities collected by the first-type trip of shuttle f when it visits node i is limited by the available quantity $\lambda_i(t_{if}^{\text{st}} - e_i)$, but not necessarily the shuttle must collect all the available quantity.

Therefore, the MILP model for the MT-VRPIP is provided in Eq. (1)-(13) and (15)-(21).

4. Solution method

This section describes the two methods proposed to solve the MT-VRPIP. The first method is named the hybrid-ILS since it combines an ILS framework and a variable neighborhood descent (VND) algorithm. The ILS, introduced by Lourenço et al. (2003), is an effective method to solve the VRP (Toth & Vigo, 2014). The ILS generates a sequence of local optima by alternating local search and perturbation. To enhance the ILS, a VND algorithm, introduced by Mladenović & Hansen (1997), is applied as a local search component in the hybrid-ILS. The VND consists of a systematic change of neighborhood each time no improvement is achieved in the current one (Labadie et al., 2016). The pseudo-code of the hybrid-ILS is sketched in Algorithm 1. First, an initial solution S is constructed using a parallel insertion heuristic as described in Section 4.2. The solution characteristics are presented in Section 4.1. Second, the VND component, detailed in Section 4.3, is carried to obtain the best solution S^* up to that step. Third, the perturbation, presented in Section 4.4, and VND components are executed while the maximum number of iterations $maxIte$ and the maximum number of iterations without improvement $noImp$ are not achieved. Finally, Section 4.5 details the second method, which is named the hybrid-ILS+MILP since it combines the hybrid-ILS with a MILP component for local search.

4.1. Search space

A solution $S = \{\pi_1^{\text{st}}, ..., \pi_u^{\text{st}}\} \cup \{\pi_1^{\text{nd}}, ..., \pi_u^{\text{nd}}\}$ is defined as a set of u first-type trips π^{st} and u second-type trips π^{nd}. A trip $r = \langle \sigma_0^r, \sigma_1^r, ..., \sigma_{n_r}^r, \sigma_{n_r+1}^r \rangle$ with $r \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$ starts at the BC denoted as σ_0^r, visits n_r CSs, and returns to the BC denoted as $\sigma_{n_r+1}^r$. Henceforth,

Algorithm 1 Hybrid-ILS

```
1: procedure ILS(maxIte,noImp)
2:     S := ParallelInsertion()                    ▷ Procedure in Section 4.2
3:     S* := VND(S)                                ▷ Procedure in Algorithm 2
4:     i := 0, j := 0
5:     while i ≤ maxIte and j ≤ noImp do
6:         S := Perturbation(S*)                   ▷ Procedure in Section 4.4
7:         S := VND(S)
8:         i := i + 1, j := j + 1
9:         if φ(S) < φ(S*) then                    ▷ Function φ(S) in Eq. 27
10:            S* := S
11:            j := 0
12:        end if
13:    end while
14:    return S*
15: end procedure
```

σ_v^r denotes the vertex at each stop $v = \{0, ..., n_r + 1\}$ of trip r. Each CS $i \in V$ may be visited at most once by the first-type trips and at most once by the second-type trips. On the way to a node σ_v^r, its service starting time is denoted as t_v^r and its delay is given by $\beta_v^r = \max\{t_v^r - l_{\sigma_v^r}, 0\}$. By definition, the service starting time t_v^r of node σ_v^r is grater than or equal to its opening time $e_{\sigma_v^r}$. The following quantities characterize a trip $r \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$:

$$\text{Travel time } c(r) = \sum_{v=0}^{n_r} c_{\sigma_v^r, \sigma_{v+1}^r} \tag{22}$$

$$\text{Maximum delay } \delta(r) = \max\{\beta_1^r, ..., \beta_{n_r}^r\} \tag{23}$$

The cost $\phi(r)$ of trip $r \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$ is defined as its traveling cost plus its maximum delay cost plus its depot delay cost as presented in Eq. (24). Additionally, the collected WB units at node σ_v^r of trip $r \in \{\pi^{\text{st}}\}$ with $v = \{0, ..., n_r + 1\}$ is given by $y_v^r = \min\{\lambda_{\sigma_v^r}(t_v^r - e_{\sigma_v^r}), p_{\sigma_v^r}\}$. Thus, a trip $r \in \{\pi^{\text{st}}\}$ is also characterized by the quantity $y(r)$, defined in Eq. (25).

$$\phi(r) = \omega^{\text{t}} c(r) + \omega^{\text{d}}[\delta(r) + \beta_{n_r+1}^r] \tag{24}$$

$$\text{Collected WB units } y(r) = \sum_{v=1}^{n_r} y_v^r \tag{25}$$

The shortage cost $U(S)$ of solution S is given in Eq. (26), where $y^{\text{st}}(S)$ represents the WB

units collected by the first-type trips, i.e., $y^{\text{st}}(S) = \sum_{f=1}^{u} y(\pi_f^{\text{st}})$. The cost $\phi(S)$ of solution S involving a set of trips π^{st} and π^{nd} is given by Eq. (27).

$$U(S) = \omega^{\text{s}} \max\left\{q - y^{\text{st}}(S), 0\right\} \tag{26}$$

$$\phi(S) = \sum_{f=1}^{u} \phi(\pi_f^{\text{st}}) + \sum_{f=1}^{u} \phi(\pi_f^{\text{nd}}) + U(S) \tag{27}$$

4.2. Initial solution

To build the initial solution of the hybrid-ILS, two procedures are executed to create the first- and second-type trips. The procedure to create the first-type trips follows three steps. In the first step, u empty first-type trips are built, including the BC node $n+1$ with $t^{\max}$ as the service starting time. In the second step, the set V^{st} of possible nodes to be included in the first-type trips is initialized with all the CSs. In the third step, the feasibility and new cost of the solution when inserting each CS $i \in V^{\text{st}}$ at the second position of each trip, i.e., following a backward procedure, are evaluated. The feasibility evaluation verifies that the processing time limit is met and the service starting time of the inserted node is greater or equal than its opening time. The new cost of the solution is calculated according to Eq. (27). When the evaluation of the feasibility and cost of each CS at each trip is completed, the solution is updated with the feasible insertion that generates the greatest decrease in the total cost and the inserted node is removed from the set V^{st}. The third step is performed until the shortage cost of the solution equals zero, the size of the set V^{st} equals zero, all insertions increase the total cost, or all insertions are unfeasible.

The procedure to build the second-type trips follows also three steps. In the first step, u empty second-type trips are created, including the BC node 0 with a service starting time equals to $t^{\max}$. In the second step, all CSs with remaining profit to collect after building the first-type trips are aggregated to the set V^{nd}. The third step runs a set of iterations until the size of the set V^{nd} equals zero. In each iteration, the new cost of the solution when inserting each CS $i \in V^{\text{nd}}$ at the penultimate position of each trip, i.e., following a forward procedure, is evaluated according to Eq. (27). The solution is updated in each iteration with the insertion that generates the lowest increase in the total cost; and the inserted node is removed from the set V^{nd} just after.

4.3. Local search: VND

The VND procedure is included as a local search component following a best improvement strategy. The proposed VND on a solution S considering a set of neighborhoods $\mathcal{N}^{\text{st}}$ on the first-type trips and a set of neighborhoods $\mathcal{N}^{\text{nd}}$ on the second-type trips is presented in Algorithm 2. The neighborhoods are detailed in Section 4.3.1 and the change in the objective function with each neighborhood movement is explained in Section 4.3.2.

The VND for the MT-VRPIP initiates with the procedure $VNDforFirstTrips$ by exploring the first neighborhood of the set $\mathcal{N}^{\text{st}}$. If the exploration does not obtain a lower-cost solution in the current neighborhood, the next one is explored. In contrast, if the search yields a lower-cost solution, the procedure $VNDforFirstTrips$ restarts the exploration in the first neighborhood of the set $\mathcal{N}^{\text{st}}$. The procedure $VNDforFirstTrips$ continues in the same way with successive neighborhoods of the set $\mathcal{N}^{\text{st}}$ until it explores the last neighborhood and no lower-cost solutions are found. Then, the procedure $VNDforSecondTrips$ explores the set $\mathcal{N}^{\text{nd}}$ similarly to the procedure $VNDforFirstTrips$. The VND is executed while the procedures $VNDforFirstTrip$ and $VNDforSecondTrips$ improve the solution.

Algorithm 2 VND procedure on a solution S for the MT-VRPIP

procedure VND($S, \mathcal{N}^{\text{st}}, \mathcal{N}^{\text{nd}}$)
  $imp :=$ true
  while $imp =$ true **do**
    $imp :=$ false
    $\widetilde{S} :=$ VNDforFirstTrips($S, \mathcal{N}^{\text{st}}$)
    $\widetilde{S} :=$ VNDforSecondTrips($\widetilde{S}, \mathcal{N}^{\text{nd}}$)
    if $\phi(\widetilde{S}) < \phi(S)$ **then** ▷ $\phi(S)$ is presented in Eq. 27
      $S := \widetilde{S}$; $imp :=$ true
    end if
  end while
  return S
end procedure

4.3.1. Neighborhoods

The set of neighborhoods for the VND is composed of classic operators: swap, relocate, and 2-opt*. The following operators are implemented on trips $r, \tilde{r} \in \{\pi^k\}$ of type $k \in \{\text{st}, \text{nd}\}$:

- *swapIntra*: Swap nodes σ_v^r and σ_w^r at positions $v, w = \{1, ..., n_r\}$ of trip r with $v \neq w$.

- *relocateIntra*: Remove node σ_v^r from position v of trip r and reinsert it at position w of the same trip with $v, w = \{1, ..., n_r\}$ and $v \neq w$.

- *swapInter*: Swap nodes σ_v^r and $\sigma_w^{\tilde{r}}$ at positions $v = \{1, ..., n_r\}$ and $w = \{1, ..., n_{\tilde{r}}\}$ of two distinct trips r and $\tilde{r}$.

- *relocateInter*: Remove node σ_v^r from position $v = \{1, ..., n_r\}$ of trip r and reinsert it at position $w = \{1, ..., n_{\tilde{r}}\}$ of a distinct trip $\tilde{r}$.

- 2*opt**: Swap sequences $\langle \sigma_v^r, ..., \sigma_{n_r}^r \rangle$ and $\langle \sigma_w^{\tilde{r}}, ..., \sigma_{n_{\tilde{r}}}^{\tilde{r}} \rangle$ of two distinct trips r and $\tilde{r}$ with $v = \{0, ..., n_r - 1\}$ and $w = \{0, ..., n_{\tilde{r}} - 1\}$.

Additionally, three operators are proposed on the unvisited CSs by the first-type trips. For this, a dummy set $\bar{\pi}^{\text{st}} = \langle \sigma_1^{\bar{r}}, ..., \sigma_{n_{\bar{r}}}^{\bar{r}} \rangle$ with the unvisited CSs is defined. As it is a dummy set, its traveling time $c(\bar{\pi}^{\text{st}})$, maximum delay $\delta(\bar{\pi}^{\text{st}})$, and collected quantity $y(\bar{\pi}^{\text{st}})$ are set to zero. The following neighborhoods are implemented on the trips $r \in \{\pi^{\text{st}}\}$ and $\bar{r} \in \{\bar{\pi}^{\text{st}}\}$.

- *swapUnv*: Swap nodes σ_v^r and $\sigma_w^{\bar{r}}$ at positions $v = \{1, ..., n_r\}$ and $w = \{1, ..., n_{\bar{r}}\}$ of trips r and $\bar{r}$, respectively.

- *add*: Remove an unvisited node $\sigma_v^{\bar{r}}$ from position $v = \{1, ..., n_{\bar{r}}\}$ of trip $\bar{r}$ and reinsert it at position $w = \{1, ..., n_r\}$ of trip r.

- *remove*: Remove a visited node σ_v^r from position $v = \{1, ..., n_r\}$ of trip r and reinsert it at any position of trip $\bar{r}$.

Henceforth, the intra-, inter-, and unvisited-movements will refer to the set of neighborhoods $\{swapIntra, relocateIntra\}$, $\{swapInter, relocateInter, 2opt^*\}$, and $\{swapUnv, add, remove\}$, respectively. Since the unvisited CSs by the second-type trips depend on the collected quantities by the first-type trips, the VND does not consider operators on the unvisited CSs by the second-type trips. The effect of the first-type trip operators on the second-type trips and the order in which the neighborhoods are applied are discussed in Section 4.3.2 and 5.2, respectively.

4.3.2. Move evaluation

Evaluating moves in the MT-VRPIP implies to compute the change in the travel, delay, and shortage costs. Any such movement can be viewed as a separation of routes into

subsequences, which are then concatenated into new routes (Vidal et al., 2013). For instance in Fig. 3, the *swapInter* movement between two trips $r \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$ and $\tilde{r} \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$ of the same type produces indeed two new trips $\langle \sigma_0^r, ..., \sigma_{v-1}^r \rangle \oplus \langle \sigma_w^{\tilde{r}} \rangle \oplus \langle \sigma_{v+1}^r, ..., \sigma_{n_r+1}^r \rangle$ and $\langle \sigma_0^{\tilde{r}}, ..., \sigma_{w-1}^{\tilde{r}} \rangle \oplus \langle \sigma_v^r \rangle \oplus \langle \sigma_{w+1}^{\tilde{r}}, ..., \sigma_{n_{\tilde{r}}+1}^{\tilde{r}} \rangle$ where $\oplus$ represents the concatenation operator.

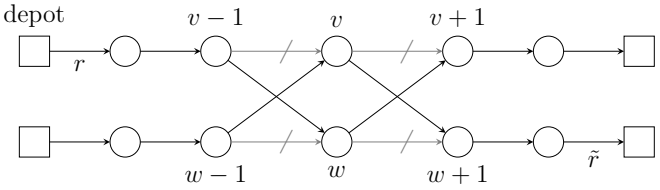

Figure 3: Swap between trips $r \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$ and $\tilde{r} \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$ of the same type.

For each subsequence σ^r of a trip $r \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$, the accumulated traveling time $C(\sigma^r)$ and the earliest $E(\sigma^r)$ and latest $L(\sigma^r)$ times to visit the first vertex are computed. $E(\sigma^r)$ and $L(\sigma^r)$ minimize the accumulated waiting times and delays of the subsequence. Moreover, the minimum duration $D(\sigma^r)$, maximum delay $B(\sigma^r)$, and accumulated waiting time $WT(\sigma^r)$ are calculated for the subsequence when starting between $E(\sigma^r)$ and $L(\sigma^r)$.

In addition, for each subsequence σ^r of a first-type trip, i.e. $r = \pi^{\text{st}}$, it is computed the earliest departure time $ES(\sigma^r)$ and the collected quantities $QE(\sigma^r)$ and $QL(\sigma^r)$ when starting the sequence at $E(\sigma^r)$ and $L(\sigma^r)$, respectively. Also, the maximum time $M(\sigma^r)$ to visit the first vertex of the sequence is computed, which avoid the minimization of the accumulated waiting times and delays but meeting the arrival time limit $t^{\max}$. Then, the duration $DM(\sigma^r)$, maximum delay $BM(\sigma^r)$, and collected quantity $QM(\sigma^r)$ are calculated for the subsequence when starting at $M(\sigma^r)$. Finally, for each subsequence σ^r of a second-type trip, i.e. $r = \pi^{\text{nd}}$, it is computed the BC delay $DD(\sigma^r)$.

Initial values for the quantities computed for a subsequence involving a single vertex σ_i^r, with $r \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$, are given by Eq. (A.1)-(A.11) in Appendix A. The data characterizing the concatenation of two subsequences $\sigma^r = \langle \sigma_i^r, ..., \sigma_j^r \rangle$ and $\sigma^{\tilde{r}} = \langle \sigma_v^{\tilde{r}}, ..., \sigma_w^{\tilde{r}} \rangle$ of the same type of trip with $r, \tilde{r} \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$ are computed using Eq. (B.1)-(B.14) in Appendix B.

The cost of a new solution $\widetilde{S}$ that modify an initial first-type trip π^{st} from an initial solution S to get a final trip $\widetilde{\pi}^{\text{st}}$ through an intra- or unv-movement is calculated following five steps. In the first step, it is verified that trip $\tilde{r} = \widetilde{\pi}^{\text{st}}$, also represented by sequence $\sigma^{\tilde{r}}$, meets the time limits $a^{\max}$ and $t^{\max}$. For this verification of feasibility, Δ_{genE} and Δ_{genM} are

defined as the time that can elapse from $E(\sigma^{\tilde{r}})$ and $M(\sigma^{\tilde{r}})$, respectively, without exceeding $a^{\max}$ and $t^{\max}$. Δ_{genE} is calculated in Eq. (28) and Δ_{genM} is computed using Eq. (28) but changing $E(\sigma^{\tilde{r}})$ and $D(\sigma^{\tilde{r}})$ for $M(\sigma^{\tilde{r}})$ and $DM(\sigma^{\tilde{r}})$, respectively. The trip $\widetilde{\pi}^{\text{st}}$ is feasible if $\Delta_{\text{genE}} \geq 0$ or $\Delta_{\text{genM}} \geq 0$. If $\widetilde{\pi}^{\text{st}}$ is feasible, the remaining steps are executed. In contrast, if $\widetilde{\pi}^{\text{st}}$ is infeasible, it is discarded and another movement is applied on the initial trip π^{st}.

$$\Delta_{\text{genE}} = \min\{a^{\max} - E(\sigma^{\tilde{r}}) - D(\sigma^{\tilde{r}}) + s_{n+1} + ES(\sigma^{\tilde{r}}), t^{\max} - E(\sigma^{\tilde{r}}) - D(\sigma^{\tilde{r}}) + s_{n+1}\} \quad (28)$$

In the second step, it is determined the set $T^{\tilde{r}}$ of possible departure times for the new trip $\tilde{r} = \widetilde{\pi}^{\text{st}}$. If $\Delta_{\text{genE}} = 0$, only the time $E(\sigma^{\tilde{r}})$ is added to the set $T^{\tilde{r}}$. In contrast, if $\Delta_{\text{genE}} > 0$, the following four times are added to the set $T^{\tilde{r}}$, but avoiding the repetition of times within the set. First, $L(\sigma^{\tilde{r}} \oplus \sigma^{r})$ if it is less or equal to $E(\sigma^{\tilde{r}}) + \Delta_{\text{genE}}$. $L(\sigma^{\tilde{r}} \oplus \sigma^{r})$ is calculated using Eq. (B.3) and represents the latest time to visit the first vertex of first-type trip $\sigma^{\tilde{r}}$ when this trip and its second-type trip σ^{r} are concatenated. Second, $L(\sigma^{\tilde{r}})$ if it is less than or equal to $E(\sigma^{\tilde{r}}) + \Delta_{\text{genE}}$. Third, $E(\sigma^{\tilde{r}}) + \Delta_{\text{genE}}$ if it is less than $L(\sigma^{\tilde{r}})$. Fourth, the time t^{q} if (i) the collected quantity with the earliest element of the set $T^{\tilde{r}}$ is less than the remaining demand to supply q^{rem}, and (ii) the collected quantity of the latest element is greater than q^{rem}. Then, when starting the trip $\widetilde{\pi}^{\text{st}}$ at time t^{q}, it collects a quantity equals to q^{rem}, which is calculated as $q^{\text{rem}} = \max\{0, q - y^{\text{st}}(S) + y(\pi^{\text{st}})\}$. Finally, if $\Delta_{\text{genM}} \geq 0$ and $M(\sigma^{\tilde{r}})$ is different to all the possible departure times mentioned before, $M(\sigma^{\tilde{r}})$ is added to the set $T^{\tilde{r}}$.

In the third step, the change in the cost of the initial solution S is calculated for each time $t \in T^{\tilde{r}}$. An intra- or unv-movement on a first-type trip generate a change in (i) the cost of the trip, (ii) the shortage cost, and (iii) the cost of the second-type trips. The change on the cost of the trip π^{st} and on the shortage cost of the initial solution S when applying an intra- or unv-movement on the trip π^{st} and starting the resulting trip $\widetilde{\pi}^{\text{st}}$ at time $t \in T^{\tilde{r}}$ is calculated using Eq. (29) and (30), respectively. In Eq. (30), $y_t(\widetilde{\pi}^{\text{st}})$ is defined as the collected quantity in the trip $\widetilde{\pi}^{\text{st}}$ when starting at time t. If $t = E(\sigma^{\tilde{r}})$, $t = L(\sigma^{\tilde{r}})$, or $t = M(\sigma^{\tilde{r}})$, $y_t(\widetilde{\pi}^{\text{st}})$ equals $QE(\sigma^{\tilde{r}})$, $QL(\sigma^{\tilde{r}})$, or $QM(\sigma^{\tilde{r}})$, respectively. If t equals another

time, $y_t(\widetilde{\pi}^{\mathrm{st}})$ is calculated in a complexity $\mathcal{O}(n)$.

$$\Delta^t_{\phi(\pi^{\mathrm{st}})} = \begin{cases} \omega^{\mathrm{t}}[c(\pi^{\mathrm{st}}) - C(\sigma^{\tilde{r}})] + \omega^{\mathrm{d}}[\beta(\pi^{\mathrm{st}}) - BM(\sigma^{\tilde{r}})] & \text{if } t = M(\sigma^{\tilde{r}}) \\ \omega^{\mathrm{t}}[c(\pi^{\mathrm{st}}) - C(\sigma^{\tilde{r}})] + \omega^{\mathrm{d}}[\beta(\pi^{\mathrm{st}}) - B(\sigma^{\tilde{r}})] & \text{otherwise} \end{cases} \tag{29}$$

$$\Delta^t_{U(S)} = U(S) - \omega^{\mathrm{s}}[\max\{0, q - y^{\mathrm{st}}(S) + y(\pi^{\mathrm{st}}) - y_t(\widetilde{\pi}^{\mathrm{st}})\}] \tag{30}$$

The change on the cost of the second-type trips by applying an intra- or unv-movement on the first-type trip π^{st} and starting the resulting trip $\widetilde{\pi}^{\mathrm{st}}$ at time $t \in T^{\tilde{r}}$ is calculated as follows. First, it is determined the set LN of CSs visited at their closing times by the initial trip π^{st} and the set $\widetilde{LN}$ of CSs visited at their closing times when starting the new trip $\widetilde{\pi}^{\mathrm{st}}$ at time $t \in T^{\tilde{r}}$. Second, the elements of sets LN and $\widetilde{LN}$ are compared. If a node in LN does not belong to $\widetilde{LN}$, this node must be randomly added to a second-type trip. If a node in $\widetilde{LN}$ does not belong to LN, this node must be removed from its second-type trip. Then, the second-type trips affected by the comparison of LN and $\widetilde{LN}$ form the set Π. Third, the change on the cost $\Delta^t_{\phi(r)}$ of the initial second-type trip $r = \pi^{\mathrm{nd}}$, which becomes trip $\hat{\pi}^{\mathrm{nd}} \in \Pi$ when starting the new first-type trip $\widetilde{\pi}^{\mathrm{st}}$ at time $t \in T^{\tilde{r}}$ is calculated using Eq. (31). The sequences σ^r and $\sigma^{\hat{r}}$ are used to represent the trips π^{nd} and $\hat{\pi}^{\mathrm{nd}}$, respectively.

$$\begin{aligned} \Delta^t_{\phi(r)} = \omega^{\mathrm{t}}[c(r) - C(\sigma^{\hat{r}})] + \omega^{\mathrm{d}}[\delta(r) - B_{\ddot{t}}(\sigma^{\hat{r}})] + \\ \omega^{\mathrm{d}}[\beta^r_{n_r+1} - \max\{0, \ddot{t}^{\mathrm{end}} + D_{\ddot{t}}(\sigma^{\hat{r}}) - s_{n+1} - l_{n+1}\}] \end{aligned} \tag{31}$$

In Eq. (31), $\ddot{t}^{\mathrm{end}}$ represents the ending time of the first-type trip $\ddot{r} = \ddot{\pi}^{\mathrm{st}}$, which precedes the second-type trip $\hat{\pi}^{\mathrm{nd}}$. Quantities $B_{\ddot{t}}(\sigma^{\hat{r}})$ and $D_{\ddot{t}}(\sigma^{\hat{r}})$ refer to the maximum delay and accumulated duration, respectively, of sequence $\sigma^{\hat{r}}$ when the first vertex of the first-type trip $\ddot{\pi}^{\mathrm{st}}$ is visited at time $\ddot{t}$. The quantities $\ddot{t}^{\mathrm{end}}$, $B_{\ddot{t}}(\sigma^{\hat{r}})$, and $D_{\ddot{t}}(\sigma^{\hat{r}})$ are calculated as in Eq. (C.1)-(C.3) of Appendix C.

Finally, the change on the cost of second-type trips $\Delta_{\phi(\Pi)}$ is calculated as follows:

$$\Delta^t_{\phi(\Pi)} = \sum_{f \in \Pi} \Delta^t_{\phi(f)} \tag{32}$$

Therefore, the change in the cost of the initial solution S when starting the new first-type trip $\widetilde{\pi}^{\text{st}}$ at time $t \in T^{\tilde{r}}$ is calculated as follows:

$$\Delta^{t}_{\phi(S)} = \Delta^{t}_{\phi(\pi^{\text{st}})} + \Delta^{t}_{U(S)} + \Delta^{t}_{\phi(\Pi)} \quad t \in T^{\tilde{r}} \tag{33}$$

In the fourth step, the time $t \in T^{\tilde{r}}$ that generates the greatest-positive change is selected as the departure time of the new trip $\widetilde{\pi}^{\text{st}}$ and is denoted as t^{best}. If no positive changes are found, the new trip is discarded and another movement is applied on the initial trip π^{st}.

In the fifth step, the cost of the new solution $\widetilde{S}$ when applying an intra- or unv-movement on the initial trip π^{st} to get a new trip $\widetilde{\pi}^{\text{st}}$ is calculated as follows:

$$\phi(\widetilde{S}) = \phi(S) - \Delta^{t^{\text{best}}}_{\phi(S)} \tag{34}$$

The cost of a new solution $\widetilde{S}$ that modify two initial first-type trips π^{st}_v and π^{st}_w to get two final trips $\widetilde{\pi}^{\text{st}}_v$ and $\widetilde{\pi}^{\text{st}}_w$ through an inter-movement is calculated following the five-step procedure to compute the cost of a solution when applying an intra-movement, but with some modifications. In the first step, the feasibility of both trips $\widetilde{\pi}^{\text{st}}_v$ and $\widetilde{\pi}^{\text{st}}_w$ is verified using Δ_{genE} and Δ_{genM} as in the previous procedure. In the second step, two sets $T^{\tilde{r}_v}$ and $T^{\tilde{r}_w}$ of possible departure times are determined for each trip $\widetilde{\pi}^{\text{st}}_v$ and $\widetilde{\pi}^{\text{st}}_w$, respectively. These sets are determined as in the second step of the previous procedure but assuming that $q^{\text{rem}} = q - y^{\text{st}}(S) + y(\pi^{\text{st}}_v) + y(\pi^{\text{st}}_w)$. In the third step, the change in the cost of the initial solution S if the new trips $\widetilde{\pi}^{\text{st}}_v$ and $\widetilde{\pi}^{\text{st}}_w$ start at times $t \in T^{\tilde{r}_v}$ and $\tilde{t} \in T^{\tilde{r}_w}$, respectively, is calculated as:

$$\Delta^{t,\tilde{t}}_{\phi(S)} = \Delta^{t}_{\phi(\pi^{\text{st}}_v)} + \Delta^{\tilde{t}}_{\phi(\pi^{\text{st}}_w)} + \Delta^{t,\tilde{t}}_{U(S)} + \Delta^{t,\tilde{t}}_{\phi(\Pi)} \quad t \in T^{\tilde{r}_v}, \tilde{t} \in T^{\tilde{r}_v} \tag{35}$$

where $\Delta^{t}_{\phi(\pi^{\text{st}}_v)}$ and $\Delta^{\tilde{t}}_{\phi(\pi^{\text{st}}_w)}$ are calculated using Eq. (29) and $\Delta^{t,\tilde{t}}_{U(S)}$ is calculated as follows:

$$\Delta^{t,\tilde{t}}_{U(S)} = U(S) - \omega^{\text{s}}[\max\{0, q - y^{\text{st}}(S) + y(\pi^{\text{st}}_v) + y(\pi^{\text{st}}_w) - y_t(\widetilde{\pi}^{\text{st}}_v) - y_t(\widetilde{\pi}^{\text{st}}_w)\}] \tag{36}$$

The term $\Delta^{t,\tilde{t}}_{\phi(\Pi)}$ in Eq. (35) is computed by following the four steps to calculate Eq. (32),

but modifying the sets LN and $\widetilde{LN}$. Here, LN groups the visited CSs at their closing times by both trips π_v^{st} and π_w^{st} and the set $\widetilde{LN}$ groups the visited CSs at their closing times by the trips $\widetilde{\pi}_v^{\text{st}}$ and $\widetilde{\pi}_w^{\text{st}}$ when starting at the times $t \in T^{\tilde{r}_v}$ and $\tilde{t} \in T^{\tilde{r}_w}$, respectively. The fourth step is the same as in the previous procedure but selecting the combination between times $t \in T^{\tilde{r}_v}$ and $\tilde{t} \in T^{\tilde{r}_w}$ that generates the greatest-positive change denoted as $(t, \tilde{t})^{\text{best}}$. In the fifth step, the cost of a new solution $\widetilde{S}$ when an inter-movement is applied on two initial trips π_v^{st} and π_w^{st} to obtain two trips $\widetilde{\pi}_v^{\text{st}}$ and $\widetilde{\pi}_w^{\text{st}}$ is calculated using the Eq. (34) but changing the term $\Delta_{\phi(S)}^{t^{\text{best}}}$ by the change in the cost $\Delta_{\phi(S)}^{(t,\tilde{t})^{\text{best}}}$.

The change in the cost of a second-type trip $r = \pi^{\text{nd}}$ when applying an intra-, unv- or inter-movement to get a new second-type trip $\hat{\pi}^{\text{nd}}$ is calculated as follows:

$$\Delta_{\phi(r)} = \omega^{\text{t}}[c(r) - C(\sigma^{\hat{r}})] + \omega^{\text{d}}[\delta(r) - B_{\ddot{t}}(\sigma^{\hat{r}})] + \omega^{\text{d}}[\beta_{n_r+1}^{r} - \max\{0, t_{n_{\ddot{r}+1}}^{\ddot{r}} + D_{\ddot{t}}(\sigma^{\hat{r}}) - s_{n+1} - l_{n+1}\}] \tag{37}$$

where $\sigma^{\hat{r}}$ represents the new trip $\hat{\pi}^{\text{nd}}$ and $t_{n_{\ddot{r}+1}}^{\ddot{r}}$ represents the ending time of the first-type trip $\ddot{r} = \pi^{\text{st}}$ before the new second-type trip $\hat{\pi}^{\text{nd}}$. Additionally, $B_{\ddot{t}}(\sigma^{\hat{r}})$ and $D_{\ddot{t}}(\sigma^{\hat{r}})$ refer to the maximum delay and accumulated duration, respectively, of sequence $\sigma^{\hat{r}}$ when the first vertex of the first-type trip $\ddot{r} = \pi^{\text{st}}$ is visited at time $\ddot{t} = t_0^{\ddot{r}}$. $B_{\ddot{t}}(\sigma^{\hat{r}})$ and $D_{\ddot{t}}(\sigma^{\hat{r}})$ are calculated using Eq. (C.2) and (C.3) but changing $\ddot{t}^{\text{end}}$ by $t_{n_{\ddot{r}+1}}^{\ddot{r}}$. Therefore, the cost of a new solution $\widetilde{S}$ when applying an intra-movement on a second-type trip is calculated using the Eq. (34) but changing the term $\Delta_{\phi(S)}^{t^{\text{best}}}$ by the change in the cost $\Delta_{\phi(r)}$ of the trip $r = \pi^{\text{nd}}$. Additionally, the cost of a new solution $\widetilde{S}$ when applying an inter-movement on two second-type trips π_v^{nd} and π_w^{nd} is calculated using the Eq. (34), but changing the term $\Delta_{\phi(S)}^{t^{\text{best}}}$ by the change in the costs $\Delta_{\phi(r)}$ and $\Delta_{\phi(r)}$ of both trips, respectively.

4.4. Perturbation

The perturbation used in the hybrid-ILS consists of three procedures. The first procedure modifies a fraction of nodes in the first-type trips by following three steps. First, the μ percentage of CSs in the first-type trips is removed from these trips. Second, the ξ percentage of CSs in the first-type trips is removed from these trips and added to the set V^{rem}. Third, a number of iterations are executed until the size of the set V^{rem} equals zero. In each iteration,

a CS is randomly removed from the set V^{rem} and inserted in a random feasible position of the first-type trips. The feasibility evaluation verifies that the insertion of the selected CS in the random position meets the processing time and arrival time limits. Conversely, if all positions are infeasible, the selected CS is added to the dummy set $\bar{\pi}^{\text{st}}$.

The second procedure locates and fixes in that location one unvisited CS in a first-type trip the λ percentage of the iterations of the hybrid-ILS with the following three steps. First, the set V^{unv} is loaded with all the unvisited CSs of the set $\bar{\pi}^{\text{st}}$. Second, a random number is generated. If the random number is less than λ, the third step executes a number of iterations until one unvisited CS has been fixed or the size of the set V^{unv} equals zero. In each iteration, one random CS is removed from the set V^{unv} and located in a random feasible position of the first-type trips. The feasibility evaluation verifies that the insertion meets the processing time and the arrival time limits. The located CS is not allowed to be removed from the first-type trips until the next perturbation step of the hybrid-ILS.

Finally, the third procedure randomly relocates in a different position the $\mu+\xi$ percentage of CSs in the second-type trips.

4.5. Hybrid-ILS+MILP

As exposed in Section 4.3.2, the hybrid-ILS selects the departure times of the first-type trips directly involved in a movement from a set of possible departure times. However, applying a movement on one or two routes may imply a change in the departure times not only of the involved trips but also of the other first-type trips. Therefore, a new component is implemented in the hybrid-ILS to evaluate the movements on first-type trips through a MILP. This variation is called the hybrid-ILS+MILP.

The proposed MILP for the hybrid-ILS+MILP decides the service starting time at node i by trip type k of shuttle f, denoted by real variables t^k_{if}. These decisions set the collected WB units and the delay at node i by type trip k if shuttle f, which are denoted by integer variables y^k_{if} and real variables β^k_{if}, respectively. Additionally, the visit time decisions set the units of unsatisfied demand, noted by real variable d, and the maximum delay of shuttle f in trip type k considering delays on CSs, denoted by real variables δ^k_f. The proposed MILP for the hybrid-ILS+MILP seeks to minimize the delay and shortage costs as exposed in Eq.

(38) subject to constraints in Eq. (6), (9), (11), (13)-(17), (19), (20), and (39)-(43). The Eq. (6), (9), (11), (13)-(17), (19), and (20) are used but decision variables x_{ijf}^k are replaced with the parameters $\widetilde{x}_{ijf}^k$, which set the first- and second-type trip. $\widetilde{x}_{ijf}^k$ takes a value of 1 if the arc $(i,j) \in A$ is traversed by the trip of type $k \in K$ of shuttle $f \in F$. The constraints in Eq. (39)-(43) are homologous to the Eq. (5), (7), (8), (10), (12), respectively.

$$\text{Min} \quad \omega^{\text{d}} \sum_{f \in F} \sum_{k \in K} \left(\beta_{(n+1)f}^k + \delta_f^k \right) + \omega^{\text{s}} d \tag{38}$$

$$\widetilde{x}_{ijf}^k \left(t_{if}^k + s_i + c_{ij} \right) \leq t_{jf}^k \quad (i,j) \in A; f \in F; k \in K \tag{39}$$

$$t_{(n+1)f}^{\text{st}} \sum_{i=1}^{n} \widetilde{x}_{i(n+1)f}^{\text{st}} \leq t^{\max} \quad f \in F \tag{40}$$

$$\sum_{j=1}^{n+1} \widetilde{x}_{ijf}^{\text{st}} \sum_{j=1}^{n} \widetilde{x}_{j(n+1)f}^{\text{st}} \left(t_{(n+1)f}^{\text{st}} - e_i \right) \leq a^{\max} \quad i \in V; f \in F \tag{41}$$

$$t_{0f}^{\text{nd}} \geq \left(t_{(n+1)f}^{\text{st}} + s_{n+1} \right) \sum_{i=0}^{n} \widetilde{x}_{i(n+1)f}^{\text{st}} \quad f \in F \tag{42}$$

$$\sum_{j=0}^{n} \widetilde{x}_{jif}^k \left(t_{if}^k - \beta_{if}^k \right) \leq l_i \quad i \in V' \backslash \{0\}; f \in F; k \in K \tag{43}$$

The proposed MILP for the hybrid-ILS+MILP is run after it is determined that a movement generates an improvement with the method explained in Section 4.3.2.

5. Computational experiments

This section presents the results of the computational experiments performed on a set of 103 instances, which are created for the MT-VRPIP and are based on the blood collection system of Bogota, Colombia. Three solution methods are used: the optimizers in the software CPLEX V 12.8, the hybrid-ILS, and the hybrid-ILS+MILP. The methods are implemented in JAVA and are executed on a machine with an Intel(R) Core(TM) i7-3770S processor with 3.10 GHz speed and 8 GB RAM.

The process of the instance generation is detailed in Section 5.1. Then, the parameter setting of the hybrid-ILS and the hybrid-ILS+MILP is presented in Section 5.2. The results on the set of instances are provided in Section 5.3. In addition, BKSs are highlighted for the

proposed instances. Section 5.4 compares the performance of the three solution methods.

5.1. Instances

The BC of each instance is randomly selected from the set of 16 BCs located in Bogota, Colombia (Secretaria Distrital de Salud, 2017). For each BC, the profit and service time are assumed to be zero. The opening and closing times of activity for each BC are set equals to its operating hours on week days, which are obtained from its web site.

The number n of CSs for each instance is randomly selected from the set of 58 CSs reported by the blood collection system of Bogota in 2017 (Secretaria Distrital de Salud, 2017), which are shown in Fig. 4. Additionally, $n = \{5 - 15, 20, 25, 30, 35, 40, 45, 50, 58\}$. The service time of each CS is generated following a discrete uniform distribution between 5 and 20 min. Only 48 CSs reported their donation level or profit and only 14 included their time intervals of activity. Therefore, the remaining values are randomly generated.

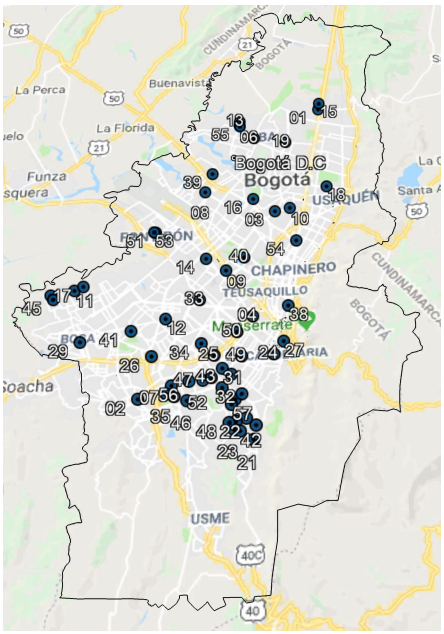

Figure 4: Mobile CSs reported by the blood collection system of Bogota, Colombia in 2017 (Secretaria Distrital de Salud, 2017)

To generate the missing values, the 58 CSs are classified into five CS categories: malls, institutions (schools, universities, or companies), churches, public sites (parks or sites without closing time), and transportation stations. For each CS category, two characteristics are identified. The first characteristic is the distribution function that fits with the highest

p-value the reported profits. The second characteristic is the probability of an early-starting priority based on the number of CSs starting the collection activity in the morning. Table 2 summarizes the distribution functions and the probabilities of an early-starting priority.

Table 2: Probability distributions to generate profits and time intervals of activity.

Category	Probability distribution of profits	Probability of an early-starting priority
Mall	Triangular with $a = 39$, $c = 39$, $b = 130$	0.2
Institution	Uniform with $a = 9$, $b = 58$.	0.8
Church	Triangular with $a = 19$, $c = 19$, $b = 72.6$	0.8
Public site	Exponential with $\lambda = 53.5$	0.67
Transportation station	Uniform with $a = 37$, $b = 80$	0.8

Each missing profit is generated following the distribution function of its CS category. Additionally, each missing time interval of collection activity $[e, l]$ is calculated following two steps. The first step consists of randomly assign an early-starting priority according the probabilities of its CS category as exposed in Table 2. In the second step, the missing time interval of activity is calculated using the following equations:

$$e = \begin{cases} \max\{420, e^{\mathrm{r}} + 60\} & \text{if the CS has an early-starting priority} \\ \max\{\min\{1140, l^{\mathrm{r}}\} - dur, e^{\mathrm{r}} + 60\} & \text{otherwise} \end{cases} \tag{44}$$

$$l = \begin{cases} \min\{\max\{420, e^{\mathrm{r}} + 60\} + dur, l^{\mathrm{r}}\} & \text{if the CS has an early-starting priority} \\ \min\{1140, l^{\mathrm{r}}\} & \text{otherwise} \end{cases} \tag{45}$$

In Eq. (44) and (45), variable dur represents the collection journey duration (in min) and was randomly selected from the set $\{360, 390, 420, 450, 480\}$. Variables e^{r} and l^{r} represent the opening and the closing (in min), respectively, of the node where the CS was located and were consulted in the web site of the node. The values 420, 1140, and 60 represent the earliest minute to start the journey (7 am), the latest minute to end the journey (7 pm) and the number of minutes to set-up, respectively.

For each value of n, a set of values for the number of shuttles u is calculated. Initially, the maximum number of shuttles $uMax$ is obtained by rounding up the quotient between n and the minimum number of CSs visited per shuttle $nMin$. Considering that the minimum value

of n is 5, $nMin$ is randomly set between 1 and 5 as 3.3 CSs. Second, the natural numbers from 1 to min $\{5, uMax\}$ (both included) are added to the set of values for u. Then, if $uMax$ is greater than 5, this set is completed with the natural numbers from 5 to $uMax$ ($uMax$ included) in increments of two. For example, the set of values for u when n equals 45 is $u = \{1 - 5, 7, 9, 11, 13, 14\}$ since $uMax = 45/3.3 = 13.64 \approx 14$.

A set of 103 instances is created, which comprises the combination of each possible value for n with each possible value for u. For each instance, the arrival time limit $t^{\max}$ is set randomly between the 50% and 60% of the time interval of activity of the BC, and the demand q is randomly defined between 30% and 40% from the total sum of profits of all CSs in the instance. For all instances, the travel times between each pair of nodes are obtained with the API Google Maps assuming a symmetric matrix. Additionally, the processing time limit $a^{\max}$, the traveling cost ω^{t}, the shortage cost ω^{s}, and the delay cost ω^{d} are fixed to 480 min, 5 USD/min, 500 USD/WB unit, and 10 USD/min, respectively, based on the BSC literature. Finally, the set of 103 instances is presented in Appendix D.

5.2. Parameter setting

The hybrid-ILS requires seven parameters: μ, ξ, λ, the orders to execute the movements on first- and second-type trips, $maxIte$, and $noImp$. The parameters μ and ξ are fixed by assuming that $\mu + \xi = \lambda$. This assumption keeps a balance between the percentage of visited nodes to modify ($\mu + \xi$) and the percentage of permutations that modify the unvisited nodes (λ). Then, it is assumed that $\mu = 0.3(\mu + \xi)$, which means that 30% of the number of nodes to modify will be deleted while the 70% will be relocated. Additionally, it is assumed that (i) the intra-movements are executed before the inter-movements since the complexity of the former is less compared with the latter and that (ii) the order within the intra- and inter-movements is randomly fixed. Therefore, *swapIntra-relocateIntra-2opt*-swapInter-relocateInter* is the order to execute the movements on second-type trips.

A 2-factor experiment is designed to investigate the effect of two factors on the performance of the hybrid-ILS and set the remaining parameters λ, the order to execute the movements on first-type trips, $maxIte$, and $noImp$. Interested readers are referred to the work of Montgomery & Runger (2010) for details on factorial experiments. The first

factor is λ with three levels: 0.15, 0.20, and 0.25. The second factor is the order to execute the movements on first-type trips with three levels: unv-inter-intra, intra-unv-inter, and intra-inter-unv. The terms *intra*, *inter*, and *unv* will be used in this section to denote the sets of intra-, inter-, and unvisited-movements, respectively. The unv-movements are randomly ordered as *add-remove-swapUnv*. As result, the experiment presents nine treatments or possible combinations between the levels of the factors.

For each treatment, five replicates are tested. Each replica consists of running the hybrid-ILS for a limited time on a random subset of 15 instances, which represent the 15% of the total set of instances. A single-factor experiment is designed to determine which time limit should be used to provide the maximum average gap with the CPLEX objective function. Five limits are considered: 900, 1200, 1500, 1800, and 7200 s. For details on this single-factor experiment, the reader is referred to Appendix E.1. As conclusion, the time limits between 1200 and 7200 s produce approximately the same average gap. Therefore, each replica of the 2-factor experiment is limited to 1200 s.

The criteria to evaluate the performance of the hybrid-ILS in each replicate of the 2-factor experiment are: (i) the average gap Δf_{CPLEX} with the objective function of CPLEX, (ii) the average iteration $\overline{iteBest}$ where the best solution is found, and (iii) the average number of iterations $\overline{nIteNo}$ between the best solution and the second best solution. The data for the three criteria of each treatment are presented in Table 3.

The normality, homoscedasticity, and independence of the experiment data residuals are verified by applying the Shapiro-Wilk, Levene, and Durbin-Watson tests. Details of the significance levels used in the tests of the 2-factor experiment and the p-values obtained with each test are presented in Appendix E.2. Then, the experiment data are subjected to the ANOVA test. It is concluded that the λ levels and the orders to execute movements on first-type trips do not affect the average gap Δf_{CPLEX}. In contrast, these factors affect the maximum iteration $\overline{iteBest}$ where the best solution is found and the maximum number of iterations $\overline{nIteNo}$ without improvement.

Tukey's test is applied to find significant differences between the means of each pair of treatments for both $\overline{iteBest}$ and $\overline{nIteNo}$ data. For the $\overline{iteBest}$ and $\overline{nIteNo}$ data, a significant difference is found between the treatment with the lowest mean, which is underlined in Table

Table 3: Results of the 2-factor experiment.

		Intra-inter-unv			Intra-unv-inter			Unv-intra-inter		
λ	Rep.	Δf_{CPLEX}	$\overline{iteBest}$	$\overline{nIteNo}$	Δf_{CPLEX}	$\overline{iteBest}$	$\overline{nIteNo}$	Δf_{CPLEX}	$\overline{iteBest}$	$\overline{nIteNo}$
0.15	1	-35.55%	1096	810	-35.55%	1606	1206	-35.54%	784	408
	2	-35.55%	1387	1051	-35.55%	1080	772	-35.55%	732	320
	3	-35.54%	1024	691	-35.55%	1465	734	-35.55%	358	198
	4	-35.54%	921	551	-35.55%	1241	765	-35.55%	922	614
	5	-35.55%	1228	821	-35.55%	1474	1044	-35.55%	537	349
	Avg.	**-35.55%**	***1131***	***785***	**-35.55%**	***1373***	***904***	**-35.55%**	**667**	**378**
0.2	1	-35.54%	603	341	-35.55%	1095	724	-35.55%	870	641
	2	-35.54%	858	489	-35.55%	1213	816	-35.54%	1283	906
	3	-35.54%	1128	633	-35.55%	1269	1087	-35.55%	1119	707
	4	-35.54%	1183	568	-35.55%	1367	913	-35.54%	790	466
	5	-35.54%	1303	759	-35.54%	1344	803	-35.54%	1361	929
	Avg.	**-35.54%**	**1015**	**558**	**-35.55%**	***1258***	***869***	**-35.54%**	**1085**	**730**
0.25	1	-35.54%	995	670	-35.55%	1306	871	-35.54%	1109	762
	2	-35.54%	1234	895	-35.55%	983	627	-35.54%	1267	988
	3	-35.54%	1244	810	-35.54%	654	406	-35.54%	1385	1132
	4	-35.54%	1080	698	-35.54%	998	540	-35.55%	1282	846
	5	-35.53%	853	559	-35.54%	574	370	-35.54%	975	621
	Avg.	**-35.54%**	**1081**	**726**	**-35.54%**	**903**	**563**	**-35.54%**	***1204***	***870***

Rep: replicate, Avg: average.

3, and the four treatments with the highest means, which are highlighted in italics in Table 3. It is concluded that the combination *unv-intra-inter* with λ equals to 0.15 gives the same average gap Δf_{CPLEX} with a smaller number of iterations.

For the hybrid-ILS and the hybrid-ILS+MILP, the parameters λ and the order to execute the movements on first-type trips are set equal to 0.15 and *unv-intra-inter*, respectively. For the hybrid-ILS, *maxIte* and *noImp* are set equal to 667 and 338, respectively. The hybrid-ILS+MILP is run the same time than the hybrid-ILS to compare the results of both methods.

5.3. Results on the set of MT-VRPIP instances

The results are grouped into four categories: small, medium, large, and hard instances. *Small*, *medium*, and *large* instances comprise instances with 5 to 11, 12 to 35, and 40 to 58 CSs, respectively. Initially, CPLEX is run with a preference for optimality and a limit of 2 h of computation. Next, if a null solution is obtained for any instance, the procedure is repeated setting the preference from optimality to feasibility. *Hard* instances include instances that could not be solved with CPLEX with either optimization priority or feasibility priority

within 2 h of computation. From the set of 103 instances, 17, 45, and 36 are small, medium, and large instances, respectively, and 5 instances belong to the hard group of instances. The hybrid-ILS and the hybrid-ILS+MILP are executed 10 for each instance.

Tables 4, 5, 6, and 7 present for each group of instances, respectively, the average cost (ϕ) of the solution, the average gap (Δ_{BKS}) to the BKS, and the average time (t), which are obtained using the three solution methods. Twelve small instances obtain optimal solution with CPLEX, while the remaining five obtain feasible solutions. The 12 optimal solutions are also reached by the hybrid-ILS and hybrid-ILS+MILP. Additionally, 40 medium instances obtain a feasible solution with an optimality preference and five with a feasibility preference. Finally, a feasible solution is obtained for 27 large instances by setting optimality preference and for nine large instances with a feasibility preference.

Table 4: Results on the small MT-VRPIP instances

Instance		Pref.	BKS	CPLEX			Hybrid-ILS			Hybrid-ILS+MILP		
n	u			ϕ	Δ_{BKS}	t	ϕ	Δ_{BKS}	t	ϕ	Δ_{BKS}	t
5	1	opt	13954.68*	13954.68	0.00%	1.32	13954.68	0.00%	1.15	13954.68	0.00%	1.16
5	2	opt	1873.00*	1873.00	0.00%	1.31	1873.00	0.00%	1.37	1873.00	0.00%	1.38
6	1	opt	1587.85*	1587.85	0.00%	0.56	1587.85	0.00%	1.31	1587.85	0.00%	1.32
6	2	opt	8433.33*	8433.33	0.00%	117.07	8433.33	0.00%	0.99	8433.33	0.00%	0.99
7	1	opt	4774.18*	4774.18	0.00%	3.04	4774.18	0.00%	0.92	4774.18	0.00%	0.92
7	2	opt	3358.75*	3358.75	0.00%	173.61	3358.75	0.00%	0.98	3358.75	0.00%	0.98
8	1	opt	34874.84*	34874.84	0.00%	215.97	34874.84	0.00%	1.07	34874.84	0.00%	1.07
8	2	opt	4368.66*	4368.66	0.00%	3383.31	4368.66	0.00%	1.59	4368.66	0.00%	1.59
9	1	opt	2957.51*	2957.51	0.00%	19.22	2957.51	0.00%	1.14	2957.51	0.00%	1.14
9	2	opt	2261.51*	2261.51	0.00%	84.95	2261.51	0.00%	2.79	2261.51	0.00%	2.79
9	3	opt	2220.51*	2220.51	0.00%	481.97	2220.51	0.00%	1.77	2220.51	0.00%	1.77
10	1	opt	3938.41	3938.41	0.00%	7200.00	3938.41	0.00%	3.40	3938.41	0.00%	3.41
10	2	opt	33311.41	33311.41	0.00%	7200.00	33311.41	0.00%	2.48	33311.41	0.00%	2.48
10	3	opt	5156.10	5156.10	0.00%	7200.00	5156.10	0.00%	1.77	5225.60	1.35%	1.80
11	1	opt	25271.33	25271.33	0.00%	7200.00	25271.33	0.00%	2.70	25271.33	0.00%	2.71
11	2	opt	37641.34	37652.16	0.03%	7200.00	37641.34	0.00%	2.34	37641.34	0.00%	2.36
11	3	opt	2388.75*	2388.75	0.00%	5204.56	2388.75	0.00%	1.75	2388.75	0.00%	1.82

BKS in USD, ϕ: average cost of the solution in USD, Δ_{BKS}: average gap to the BKS, t: average run-time in seconds, n: number of CSs, u: number of shuttles, and *opt*: optimality preference. Optimal solutions are marked with *.

5.4. Comparison of methods

Table 8 summarizes the results obtained with CPLEX, hybrid-ILS, and hybrid-ILS+MILP. This table presents the percentage of BKSs found, the average gap with respect to the BKSs (Δ_{BKS}) and the average run-time (t) of each method according to each group of instances and the entire set of instances.

Table 5: Results on the medium MT-VRPIP instances

Instance		Pref.	BKS	CPLEX			Hybrid-ILS			Hybrid-ILS+MILP		
n	u			ϕ	Δ_{BKS}	t	ϕ	Δ_{BKS}	t	ϕ	Δ_{BKS}	t
12	1	opt	37574.59	38005.92	1.15%	7200.00	37574.59	0.00%	2.25	37574.59	0.00%	2.25
12	2	opt	4176.50	4176.50	0.00%	7200.00	4176.50	0.00%	3.29	4282.47	2.54%	3.31
12	3	opt	2015.68	2015.68	0.00%	7200.00	2015.68	0.00%	2.77	2015.68	0.00%	2.78
12	4	opt	37114.41	37449.99	0.90%	7200.00	37114.41	0.00%	2.41	38000.44	2.39%	2.43
13	1	opt	83477.76	83477.76	0.00%	7200.00	83477.76	0.00%	2.31	83477.76	0.00%	2.31
13	2	opt	30595.98	30659.17	0.21%	7200.00	30595.98	0.00%	3.44	31154.59	1.83%	3.45
13	3	opt	62153.35	62611.26	0.74%	7200.00	62153.35	0.00%	3.59	62153.35	0.00%	3.59
13	4	opt	41619.83	41676.59	0.14%	7200.00	41619.83	0.00%	3.85	41619.83	0.00%	3.85
14	1	opt	7295.49	7399.01	1.42%	7200.00	7295.57	0.00%	6.37	7295.49	0.00%	6.37
14	2	opt	37851.86	38359.27	1.34%	7200.00	37851.86	0.00%	4.24	37851.86	0.00%	4.24
14	3	opt	5597.42	5845.09	4.42%	7200.00	5597.42	0.00%	5.65	5597.42	0.00%	5.66
14	4	opt	35599.08	35719.91	0.34%	7200.00	35599.08	0.00%	4.71	35647.71	0.14%	4.73
15	1	opt	3591.75	3591.75	0.00%	7200.00	3620.41	0.80%	12.40	3620.41	0.80%	12.41
15	2	opt	33809.43	35439.27	4.82%	7200.00	33809.43	0.00%	5.05	33809.43	0.00%	5.05
15	3	opt	95610.00	97204.34	1.67%	7200.00	95610.00	0.00%	4.04	95610.00	0.00%	4.05
15	4	opt	3655.59	3752.23	2.64%	7200.00	3699.07	1.19%	7.24	3655.59	0.00%	7.29
15	5	opt	8698.32	8745.73	0.55%	7200.00	8698.32	0.00%	5.88	8712.90	0.17%	5.88
20	1	opt	5441.28	5441.28	0.00%	7200.00	6331.14	16.35%	20.11	6331.14	16.35%	20.11
20	2	opt	7025.32	7616.58	8.42%	7200.00	7025.32	0.00%	19.04	7038.82	0.19%	19.05
20	3	opt	142970.70	145152.94	1.53%	7200.00	142970.70	0.00%	7.85	145073.87	1.47%	7.86
20	4	opt	7441.33	7668.33	3.05%	7200.00	7441.33	0.00%	13.37	7441.33	0.00%	13.38
20	5	opt	7645.34	8017.26	4.86%	7200.00	7645.34	0.00%	23.35	7712.26	0.88%	23.35
20	6	opt	16189.48	19206.82	18.64%	7200.00	16189.48	0.00%	20.38	16221.48	0.20%	20.39
25	1	opt	65485.81	68875.22	5.18%	7200.00	65485.81	0.00%	17.08	65485.81	0.00%	17.08
25	2	opt	5857.74	9672.43	65.12%	7200.00	5857.74	0.00%	54.60	5912.68	0.94%	54.60
25	3	opt	77747.34	107916.88	38.80%	7200.00	77747.34	0.00%	18.48	77941.69	0.25%	18.48
25	4	opt	97543.24	172961.17	77.32%	7200.00	97543.24	0.00%	25.56	97543.24	0.00%	25.57
25	5	opt	113844.51	205212.45	80.26%	7200.00	113844.51	0.00%	17.55	113844.51	0.00%	17.55
25	7	opt	49634.74	65050.13	31.06%	7200.00	49634.74	0.00%	21.51	49634.82	0.00%	21.62
25	8	opt	10716.56	76333.39	612.29%	7200.00	10716.56	0.00%	23.98	10716.56	0.00%	23.99
30	1	opt	55409.90	58195.44	5.03%	7200.00	55409.90	0.00%	73.42	55409.90	0.00%	73.49
30	2	opt	108201.12	116238.72	7.43%	7200.00	108201.12	0.00%	27.83	108204.80	0.00%	27.83
30	3	opt	37367.17	126763.89	239.24%	7200.00	37367.17	0.00%	45.17	37367.17	0.00%	45.22
30	4	opt	9540.23	23849.86	149.99%	7200.00	9647.25	1.12%	66.59	9540.23	0.00%	66.86
30	5	opt	6223.58	7292.85	17.18%	7200.00	6223.58	0.00%	61.82	6333.68	1.77%	61.85
30	7	opt	10981.33	156534.14	1325.46%	1929.97	10981.33	0.00%	60.34	11266.74	2.60%	60.41
30	9	opt	60385.75	161922.79	168.15%	7200.00	60385.75	0.00%	46.41	62656.47	3.76%	46.77
35	1	opt	141893.60	160491.07	13.11%	7200.00	141893.60	0.00%	61.31	141893.60	0.00%	61.32
35	2	feas	32200.65	59599.66	85.09%	7200.00	32200.65	0.00%	47.25	32200.65	0.00%	47.27
35	3	feas	45748.58	105024.98	129.57%	7200.00	45748.58	0.00%	84.54	48315.74	5.61%	84.56
35	4	opt	5608.83	132941.26	2270.22%	937.25	5608.83	0.00%	65.59	5677.08	1.22%	66.20
35	5	feas	62151.67	163098.73	162.42%	1418.65	62151.67	0.00%	103.33	62151.67	0.00%	103.40
35	7	feas	6391.67	140806.25	2102.96%	2576.41	6391.67	0.00%	130.22	7831.63	22.53%	130.28
35	9	opt	89996.34	173613.05	92.91%	7200.00	89996.34	0.00%	50.07	89996.34	0.00%	50.30
35	11	feas	73624.07	135011.36	83.38%	7200.00	73624.07	0.00%	51.47	75033.59	1.91%	51.52

BKS in USD, ϕ: average cost of the solution in USD, Δ_{BKS}: average gap to the BKS, t: average run-time in seconds, n: number of CSs, u: number of shuttles, *opt*: optimality preference, and *feas*: feasibility preference.

For small instances, the hybrid-ILS obtains 100% of the BKSs in 0.06% of the CPLEX time, while CPLEX finds 94.12% of the BKSs. For the medium and large instances, the hybrid-ILS finds more than 90% of the BKSs while the hybrid-ILS+MILP finds less than 60% of the BKSs in the same computation time. Additionally, the gap to the BKSs of

Table 6: Results on the large MT-VRPIP instances

Instance		Pref.	BKS	CPLEX			Hybrid-ILS			Hybrid-ILS+MILP		
n	u			ϕ	Δ_{BKS}	t	ϕ	Δ_{BKS}	t	ϕ	Δ_{BKS}	t
40	1	opt	195388.99	226351.45	15.85%	7200.00	195388.99	0.00%	77.31	195388.99	0.00%	77.32
40	2	opt	212902.24	290185.83	36.30%	6099.35	212902.24	0.00%	47.37	212902.24	0.00%	47.46
40	3	feas	9092.48	26596.11	192.51%	7200.00	9092.48	0.00%	264.12	9322.92	2.53%	264.15
40	4	feas	79899.73	159430.14	99.54%	7200.00	79899.73	0.00%	100.11	81492.17	1.99%	100.39
40	5	opt	8827.76	359291.91	3970.02%	7200.00	8827.76	0.00%	97.26	8904.78	0.87%	97.29
40	7	opt	8416.17	138060.38	1540.42%	7200.00	8416.17	0.00%	98.88	9402.47	11.72%	98.97
40	9	opt	20137.81	208377.71	934.76%	3398.44	20137.81	0.00%	107.45	20550.08	2.05%	107.45
40	11	opt	6417.14	238405.66	3615.14%	2506.16	6417.14	0.00%	133.59	6482.90	1.02%	134.48
45	1	opt	88010.55	139142.09	58.10%	7200.00	88010.55	0.00%	157.25	88010.55	0.00%	157.25
45	2	opt	13187.37	186986.96	1317.92%	4115.05	13187.37	0.00%	193.82	13387.16	1.52%	193.82
45	3	opt	125695.33	317050.39	152.24%	2478.04	125695.33	0.00%	105.84	126498.52	0.64%	106.03
45	4	opt	185186.08	409815.85	121.30%	3605.15	185186.08	0.00%	145.22	186265.33	0.58%	145.22
45	5	opt	90189.57	414577.02	359.67%	5155.10	90189.57	0.00%	101.67	90189.57	0.00%	101.79
45	7	opt	19893.27	463613.24	2230.50%	7200.00	19893.27	0.00%	219.31	20380.17	2.45%	220.00
45	9	opt	6706.09	125842.04	1776.53%	4726.92	6706.09	0.00%	172.91	6874.18	2.51%	173.42
45	13	opt	227044.71	502881.92	121.49%	2195.95	227044.71	0.00%	136.08	228841.91	0.79%	136.53
45	14	feas	15117.89	424630.54	2708.79%	2161.23	15117.89	0.00%	214.49	15336.92	1.45%	214.49
50	1	opt	216788.26	275306.09	26.99%	7200.00	216788.26	0.00%	162.20	216788.26	0.00%	162.22
50	2	opt	13026.40	284139.28	2081.26%	846.27	13173.54	1.13%	248.65	13026.40	0.00%	248.65
50	4	opt	52281.74	438470.86	738.67%	3997.75	52281.74	0.00%	254.85	52281.74	0.00%	255.00
50	7	feas	8093.67	71703.85	785.92%	5582.48	8093.67	0.00%	276.10	8804.42	8.78%	276.25
50	9	feas	8309.82	534161.31	6328.07%	1959.71	8309.82	0.00%	350.92	9129.75	9.87%	351.15
50	11	feas	122245.42	412309.35	237.28%	7200.00	122245.42	0.00%	191.11	124492.33	1.84%	191.15
50	13	feas	15586.20	400965.91	2472.57%	2614.65	15586.20	0.00%	201.52	16124.60	3.45%	203.06
50	15	opt	8571.74	279882.02	3165.17%	7200.00	8571.74	0.00%	242.64	9112.24	6.31%	243.74
58	1	opt	352766.86	530521.89	50.39%	7200.00	352766.86	0.00%	247.29	352766.86	0.00%	247.30
58	2	opt	204295.55	436105.99	113.47%	7200.00	204295.55	0.00%	310.47	204295.55	0.00%	310.47
58	3	opt	10812.48	110641.53	923.28%	7200.00	10812.48	0.00%	728.30	11095.79	2.62%	728.38
58	4	opt	297779.49	545872.04	83.31%	4146.75	297779.49	0.00%	173.69	324848.61	9.09%	173.75
58	5	opt	9590.56	485768.61	4965.07%	5569.11	9645.56	0.57%	795.71	9590.56	0.00%	795.84
58	7	opt	74571.49	548985.41	636.19%	2961.77	74571.49	0.00%	264.93	77605.04	4.07%	265.12
58	9	opt	9544.31	443716.00	4549.01%	7200.00	9544.31	0.00%	594.92	9925.15	3.99%	595.03
58	11	opt	61036.81	548468.89	798.59%	2637.81	61036.81	0.00%	386.50	62306.83	2.08%	386.78
58	15	feas	10048.00	590261.57	5774.42%	4734.22	10048.00	0.00%	305.33	12951.03	28.89%	308.51
58	17	feas	10132.65	564134.28	5467.49%	5395.41	10132.65	0.00%	494.54	10430.62	2.94%	495.59
58	18	opt	8762.14	511358.96	5736.01%	5287.19	8762.14	0.00%	362.46	9612.39	9.70%	363.07

BKS in USD, ϕ: average cost of the solution in USD, Δ_{BKS}: average gap to the BKS, t: average run-time in seconds, n: number of CSs, u: number of shuttles, *opt*: optimality preference, and *feas*: feasibility preference.

Table 7: Results on the hard MT-VRPIP instances

Instance		BKS	Hybrid-ILS			Hybrid-ILS+MILP		
n	u		ϕ	Δ_{BKS}	t	ϕ	Δ_{BKS}	t
40	12	6699.72	6699.72	0.00%	134.97	7009.22	4.62%	135.24
45	11	47715.92	47715.92	0.00%	101.75	48112.89	0.83%	102.90
50	3	122589.83	122589.83	0.00%	223.78	123140.57	0.45%	223.79
50	5	178504.34	178504.34	0.00%	165.13	178504.34	0.00%	165.50
58	13	16912.77	16912.77	0.00%	399.16	17238.18	1.92%	400.91

BKS in USD, ϕ: average cost of the solution in USD, Δ_{BKS}: average gap to the BKS, t: average run-time in seconds, n: number of CSs, and u: number of shuttles.

the hybrid-ILS is less than 1% for these groups of instances while the hybrid-ILS+MILP generates gaps of 1.50% and 3.44% for medium and large instances, respectively. In the hard instances, the hybrid-ILS finds 100% of the best solutions while the hybrid-ILS+MILP finds 20% in the same computation time. However, the hybrid-ILS+MILP is on average within a gap of less than 2% of the BKSs in the hard group of instances.

Considering all instances, the hybrid-ILS and hybrid-ILS+MILP achieve gaps to the BKSs of 0.21% and 1.95%, respectively, in 2.00% of the CPLEX time, while CPLEX obtains a gap of 734.73%. In addition, the hybrid-ILS obtains 94.17% of the BKSs with a gap of less than 1% while the hybrid-ILS+MILP obtains 50.49% of the BKSs with a gap of 1.95% in the same computational time. It is concluded that the hybrid-ILS and hybrid-ILS+MILP are superior to CPLEX in both computation time and gap to the BKSs and that the hybrid-ILS outperforms the hybrid-ILS+MILP in terms of gap to the BKSs over the total set of instances.

Table 8: Summary of results for each solution method

Metric	Instances	CPLEX	Hybrid-ILS	Hybrid-ILS+MILP
% BKS	Small	94.12%	100.00%	94.12%
	Medium	11.11%	91.11%	55.56%
	Large	0.00%	94.44%	27.78%
	Hard	0.00%	100.00%	20.00%
	All	20.39%	94.17%	50.49%
Δ_{BKS}	Small	0.00%	0.00%	0.08%
	Medium	173.76%	0.43%	1.50%
	Large	1782.90%	0.05%	3.44%
	Hard	-	0.00%	1.56%
	All	734.73%	0.21%	1.95%
t	Small	44.80	0.03	0.03
	Medium	112.24	0.50	0.50
	Large	84.94	4.15	4.16
	Hard	120.00	3.42	3.43
	All	91.94	1.84	1.84

Δ_{BKS}: average gap to the BKS, t: average run-time in minutes.

Some additional metrics are presented in Table 9 to compare the hybrid-ILS with the hybrid-ILS+MILP for each group of instances and the total set of instances. The metrics included in this table are as follows. First, the number of iterations and the percentage of time spent on the VND. Second, the percentage of the VND that is employed in the evaluation

of the movements using the evaluation method described in Section 4.3.2 (noMILP), and using the evaluation method described in Section 4.5 (subMILP). Third, the number of first-type and second-type movements and the number of improvements found with each type of movements. Fourth, the percentage of improvements of the first-type movements that are found with the noMILP and subMILP evaluation methods.

Over the entire set of instances, the hybrid-ILS executes four times more iterations than the hybrid-ILS+MILP. Therefore, the hybrid-ILS runs five times more first-type and second-type movements and finds five times more improvements than the hybrid-ILS+MILP. Therefore, each iteration of the hybrid-ILS+MILP takes longer than each iteration of the hybrid-ILS due to the execution of the MILP within the subMILP evaluation method.

Even though the MILP within the subMILP evaluation method in the hybrid-ILS+MILP is executed when an improvement is found with the noMILP evaluation method, the subMILP evaluation method accounts for 76.56% of the VND time while the noMILP evaluation method accounts for 4.16% of the VND time. The increase in the time per iteration using the subMILP evaluation method allows improving the solution obtained with the noMILP evaluation method 2.24% of the times.

It is concluded that the way of evaluating the movements in the hybrid-ILS is effective compared to the inclusion of the subMILP evaluation method in the hybrid-ILS+MILP.

Table 9: Summary of metrics for the hybrid-ILS and the hybrid-ILS+MILP

	Hybrid-ILS					Hybrid-ILS+MILP				
Metric	Small	Medium	Large	Hard	All	Small	Medium	Large	Hard	All
Iterations	388.47	479.24	571.06	435.40	494.22	100.06	110.20	128.75	57.80	112.47
% time VND	92.58%	97.62%	98.94%	99.46%	97.34%	98.12%	99.28%	99.49%	99.89%	99.19%
% time noMILP	61.71%	60.46%	61.26%	55.67%	60.71%	4.11%	4.54%	4.03%	1.84%	4.16%
% time subMILP	0.00%	0.00%	0.00%	0.00%	0.00%	76.71%	75.21%	76.93%	85.61%	76.56%
First-type mov.	7690.65	16847.78	29770.03	23204.60	20161.50	1943.53	3587.13	5421.14	3013.60	3929.03
First-type imp.	945.59	3437.36	8230.14	6427.20	4846.38	245.71	731.76	1473.69	816.80	914.98
% imp. noMILP	100.00%	100.00%	100.00%	100.00%	100.00%	99.49%	97.68%	96.96%	98.43%	97.76%
% imp. subMILP	0.00%	0.00%	0.00%	0.00%	0.00%	0.51%	2.32%	3.04%	1.57%	2.24%
Second-type mov.	4788.59	14128.40	30845.58	23450.80	18882.32	1185.71	3064.02	6206.11	3238.20	3860.67
Second-type imp.	1208.18	6333.64	17934.81	13202.80	9875.92	314.00	1453.24	3888.78	1865.00	2136.46

6. Conclusions and perspectives

In this paper, the multi-trip vehicle routing problem with increasing profits (MT-VRPIP) is introduced. This new problem arises from the context of optimizing the routing of a shuttle fleet to transport whole blood (WB) units from collection sites (CSs) to a blood center (BC), while meeting two time constraints. The first one refers to the 8-h processing limit that WB units intended to produce platelets and cryoprecipitate must meet. The second one guarantees that the WB units with the 8-h processing limit arrive at the BC before a reception time limit. With this last constraint, the BC will be able to process the platelets and cryoprecipitate before the closure of the facility. An additional characteristic of the MT-VRPIP is that donations at CSs follow a linear increasing function, which allows calculating the number of WB units collected by the shuttles when they visit the CSs.

To solve the MT-VRPIP, a mixed-integer linear programming model and two solution methods, based on the iterated local search metaheuristic, are proposed. Additionally, a new set of instances based on the blood collection system of Bogota, Colombia is designed. The computational experiments on the new set of instances show that the two proposed methods are efficient since they get the best-known solutions in the 2.00% of the average time expended by the software CPLEX.

From a managerial point of view, the problem and solution methods presented in this article are tools for decision makers in the blood supply chain. Specifically, these tools allow decision-makers to (i) optimize operational decisions such as transportation decisions in the WB collection process, (ii) consider time constraints related to the production process such as the 8-h processing limit, and (iii) manage additional decisions if the optimal transportation plan generates shortages and delays in the shuttle routes. If a shortage occurs, decision makers can generate additional supply through actions such as contacting other BCs for additional WB units or increasing the capacity of the collection fleet. If delays occur, decision makers may plan the staff who must wait for the shuttles at the CSs or BC.

In future research, other donation patterns could be tested, such as the irregular pattern proposed by Özener & Ekici (2018) or a step function. Another research perspective is to explore additional decisions such as the location of CSs, the determination of the shuttle fleet size, and/or inventory decisions.

Appendix A. Initial values for a subsequence involving a single node

Initial values for a sequence involving a single vertex σ_i^r of a first- or second-type trip, i.e. $r \in \{\pi^{\mathrm{st}}, \pi^{\mathrm{nd}}\}$, are given by:

$$C(\sigma_i^r) = B(\sigma_i^r) = WT(\sigma_i^r) = 0 \qquad \sigma_i^r \in V',\ r \in \{\pi^{\mathrm{st}}, \pi^{\mathrm{nd}}\} \tag{A.1}$$

$$E(\sigma_i^r) = \begin{cases} e_{\sigma_i^r} & \text{if } \sigma_i^r \in V',\ r = \pi^{\mathrm{st}} \\ e_{\sigma_i^r} & \text{if } \sigma_i^r \in \{0, n+1\},\ r = \pi^{\mathrm{nd}} \\ l_{\sigma_i^r} & \text{if } \sigma_i^r \in V,\ r = \pi^{\mathrm{nd}} \end{cases} \tag{A.2}$$

$$L(\sigma_i^r) = \begin{cases} \min\{l_{\sigma_i^r}, t^{\max}\} & \text{if } \sigma_i^r \in V',\ r = \pi^{\mathrm{st}} \\ l_{\sigma_i^r} & \text{if } \sigma_i^r \in V',\ r = \pi^{\mathrm{nd}} \end{cases} \tag{A.3}$$

$$D(\sigma_i^r) = s_{\sigma_i^r} \qquad \sigma_i^r \in V',\ r \in \{\pi^{\mathrm{st}}, \pi^{\mathrm{nd}}\} \tag{A.4}$$

Additional initial values for a sequence involving a single vertex σ_i^r of a first-type trip, i.e. $r = \pi^{\mathrm{st}}$, are given by the following equations where $e^{\max} = \max\{e_1, ..., e_n\}$ represents the maximum earliest-opening time of the CSs.

$$ES(\sigma_i^r) = \begin{cases} e^{\max} & \text{if } \sigma_i^r \in \{0, n+1\} \\ e_{\sigma_i^r} & \text{if } \sigma_i^r \in V \end{cases} \tag{A.5}$$

$$QE(\sigma_i^r) = 0 \qquad \sigma_i^r \in V' \tag{A.6}$$

$$QL(\sigma_i^r) = QM(\sigma_i^r) = \min\{\max\{0, \lambda_{\sigma_i^r}(t^{\max} - e_{\sigma_i^r})\}, p_{\sigma_i^r}\} \qquad \sigma_i^r \in V' \tag{A.7}$$

$$M(\sigma_i^r) = t^{\max} \qquad \sigma_i^r \in V' \tag{A.8}$$

$$DM(\sigma_i^r) = s_{\sigma_i^r} \qquad \sigma_i^r \in V' \tag{A.9}$$

$$BM(\sigma_i^r) = \max\{0, t^{\max} - l_{\sigma_i^r}\} \qquad \sigma_i^r \in V' \tag{A.10}$$

The additional initial value for a subsequence involving a single vertex σ_i^r of a second-type

trip, i.e. $r = \pi^{\text{nd}}$, is given by:

$$DD(\sigma_i^r) = \max\{0, L(\sigma_i^r) - l_{n+1}\} \qquad \sigma_i^r \in V' \tag{A.11}$$

Appendix B. Values for the concatenation of two subsequences

The concatenation of two subsequences $\sigma^r = \langle \sigma_i^r, ..., \sigma_j^r \rangle$ and $\sigma^{\tilde{r}} = \langle \sigma_v^{\tilde{r}}, ..., \sigma_w^{\tilde{r}} \rangle$ of the same type of trip, i.e., $r, \tilde{r} \in \{\pi^{\text{st}}, \pi^{\text{nd}}\}$, is characterized by the following data:

$$C(\sigma^r \oplus \sigma^{\tilde{r}}) = C(\sigma^r) + C(\sigma^{\tilde{r}}) + c_{\sigma_j^r \sigma_v^{\tilde{r}}} \tag{B.1}$$

$$E(\sigma^r \oplus \sigma^{\tilde{r}}) = \max\{E(\sigma^{\tilde{r}}) - \Delta, E(\sigma^r)\} - \Delta_{\text{WT}} \tag{B.2}$$

$$L(\sigma^r \oplus \sigma^{\tilde{r}}) = \max\{\min\{L(\sigma^{\tilde{r}}) - \Delta, L(\sigma^r)\} + \Delta_\beta, E(\sigma^r \oplus \sigma^{\tilde{r}})\} \tag{B.3}$$

$$\begin{aligned} D(\sigma^r \oplus \sigma^{\tilde{r}}) = \Delta + D(\sigma^{\tilde{r}}) + \Delta_{\text{WT}} - \min\{E(\sigma^r \oplus \sigma^{\tilde{r}}) - E(\sigma^r), WT(\sigma^r)\} - \\ \min\{E(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta + \Delta_{\text{WT}} - E(\sigma^{\tilde{r}}), WT(\sigma^{\tilde{r}})\} \end{aligned} \tag{B.4}$$

$$B(\sigma^r \oplus \sigma^{\tilde{r}}) = \begin{cases} \max\{B(\sigma^r), B(\sigma^{\tilde{r}})\} & \text{if } L(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta + \Delta_{\text{WT}} \leq L(\sigma^{\tilde{r}}) \\ \max\{B(\sigma^r), B_{\text{new}}(\sigma^{\tilde{r}})\} & \text{otherwise} \end{cases} \tag{B.5}$$

$$\begin{aligned} WT(\sigma^r \oplus \sigma^{\tilde{r}}) = WT(\sigma^r) + WT(\sigma^{\tilde{r}}) + \Delta_{\text{WT}} - \min\{E(\sigma^r \oplus \sigma^{\tilde{r}}) - E(\sigma^r), WT(\sigma^r)\} - \\ \min\{E(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta + \Delta_{\text{WT}} - E(\sigma^{\tilde{r}}), WT(\sigma^{\tilde{r}})\} \end{aligned} \tag{B.6}$$

In Eq. (B.2)-(B.6) and hereafter, $\Delta = D(\sigma^r) + c_{\sigma_j^r \sigma_v^{\tilde{r}}}$, $\Delta_{\text{WT}} = \max\{E(\sigma^{\tilde{r}}) - \Delta - L(\sigma^r), 0\}$, and $\Delta_\beta = \max\{E(\sigma^r) + \Delta - L(\sigma^{\tilde{r}}), 0\}$. Additionally, in Eq. (B.5), $B_{\text{new}}(\sigma^{\tilde{r}})$ refers to the new maximum delay of subsequence $\sigma^{\tilde{r}}$ and is calculated in complexity $\mathcal{O}(n)$ starting the subsequence at time $L(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta + \Delta_{\text{WT}}$.

In addition, the concatenation of two subsequences $\sigma^r = \langle \sigma_i^r, ..., \sigma_j^r \rangle$ and $\sigma^{\tilde{r}} = \langle \sigma_v^{\tilde{r}}, ..., \sigma_w^{\tilde{r}} \rangle$

of a first-type trip, i.e. $r, \tilde{r} = \{\pi^{\text{st}}\}$, is also characterized by the following data:

$$ES(\sigma^r \oplus \sigma^{\tilde{r}}) = \min\{ES(\sigma^r), ES(\sigma^{\tilde{r}})\} \tag{B.7}$$

$$QE(\sigma^r \oplus \sigma^{\tilde{r}}) = \begin{Bmatrix} QE(\sigma^r) & \text{if } E(\sigma^r \oplus \sigma^{\tilde{r}}) = E(\sigma^r) \\ QL(\sigma^r) & \text{if } E(\sigma^r \oplus \sigma^{\tilde{r}}) = L(\sigma^r) \\ QE_{\text{new}}(\sigma^r) & \text{otherwise} \end{Bmatrix} + \tag{B.8}$$

$$\begin{Bmatrix} QE(\sigma^{\tilde{r}}) & \text{if } E(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta + \Delta_{\text{WT}} = E(\sigma^{\tilde{r}}) \\ QL(\sigma^{\tilde{r}}) & \text{if } E(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta + \Delta_{\text{WT}} = L(\sigma^{\tilde{r}}) \\ QE_{\text{new}}(\sigma^{\tilde{r}}) & \text{otherwise} \end{Bmatrix}$$

$$QL(\sigma^r \oplus \sigma^{\tilde{r}}) = \begin{Bmatrix} QE(\sigma^r) & \text{if } L(\sigma^r \oplus \sigma^{\tilde{r}}) = E(\sigma^r) \\ QL(\sigma^r) & \text{if } L(\sigma^r \oplus \sigma^{\tilde{r}}) = L(\sigma^r) \\ QL_{\text{new}}(\sigma^r) & \text{otherwise} \end{Bmatrix} + \tag{B.9}$$

$$\begin{Bmatrix} QE(\sigma^{\tilde{r}}) & \text{if } L(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta + \Delta_{\text{WT}} = E(\sigma^{\tilde{r}}) \\ QL(\sigma^{\tilde{r}}) & \text{if } L(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta + \Delta_{\text{WT}} = L(\sigma^{\tilde{r}}) \\ QL_{\text{new}}(\sigma^{\tilde{r}}) & \text{otherwise} \end{Bmatrix}$$

$$M(\sigma^r \oplus \sigma^{\tilde{r}}) = \max\{M(\sigma^{\tilde{r}}) - \Delta_{\text{M}} + \Delta_{\beta\text{M}}, E(\sigma^r \oplus \sigma^{\tilde{r}})\} \tag{B.10}$$

$$QM(\sigma^r \oplus \sigma^{\tilde{r}}) = \begin{Bmatrix} QM(\sigma^r) & \text{if } M(\sigma^r \oplus \sigma^{\tilde{r}}) = M(\sigma^r) \\ QM_{\text{new}}(\sigma^r) & \text{otherwise} \end{Bmatrix} + \tag{B.11}$$

$$\begin{Bmatrix} QM(\sigma^{\tilde{r}}) & \text{if } M(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta_{\text{M}} = M(\sigma^{\tilde{r}}) \\ QM_{\text{new}}(\sigma^{\tilde{r}}) & \text{otherwise} \end{Bmatrix}$$

$$DM(\sigma^r \oplus \sigma^{\tilde{r}}) = DM(\sigma^r) + DM(\sigma^{\tilde{r}}) + c_{\sigma_j^r \sigma_v^{\tilde{r}}} \tag{B.12}$$

$$BM(\sigma^r \oplus \sigma^{\tilde{r}}) = \begin{cases} \max\{BM(\sigma^r), BM(\sigma^{\tilde{r}}) + \Delta_{\beta\text{M}}\} & \text{if } M(\sigma^r \oplus \sigma^{\tilde{r}}) = M(\sigma^r) \\ \max\{BM_{\text{new}}(\sigma^r), BM(\sigma^{\tilde{r}}) + \Delta_{\beta\text{M}}\} & \text{otherwise} \end{cases} \tag{B.13}$$

In Eq. (B.10)-(B.13) and hereafter, $\Delta_{\text{M}} = DM(\sigma^r) + c_{\sigma_j^r \sigma_v^{\tilde{r}}}$ and $\Delta_{\beta\text{M}} = \max\{E(\sigma^r) + \Delta_{\text{M}} - M(\sigma^{\tilde{r}}), 0\}$. $QE_{\text{new}}(\sigma^r)$, $QL_{\text{new}}(\sigma^r)$, and $QM_{\text{new}}(\sigma^r)$ refer in Eq. (B.8)-(B.11) to the collected

quantity in the sequence σ^r and are calculated in $\mathcal{O}(n)$ when starting that sequence at times $E(\sigma^r \oplus \sigma^{\tilde{r}})$, $L(\sigma^r \oplus \sigma^{\tilde{r}})$, and $M(\sigma^r \oplus \sigma^{\tilde{r}})$, respectively. Additionally, $QE_{\text{new}}(\sigma^{\tilde{r}})$, $QL_{\text{new}}(\sigma^{\tilde{r}})$, and $QM_{\text{new}}(\sigma^{\tilde{r}})$ refer to the collected quantity in the sequence $\sigma^{\tilde{r}}$ and are calculated in $\mathcal{O}(n)$ when starting that sequence at times $E(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta + \Delta_{\text{WT}}$, $L(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta + \Delta_{\text{WT}}$, and $M(\sigma^r \oplus \sigma^{\tilde{r}}) + \Delta_{\text{M}}$, respectively. Finally, $BM_{\text{new}}(\sigma^r)$ refers in Eq. (B.5) to the new maximum delay of subsequence σ^r and is calculated in complexity $\mathcal{O}(n)$ starting that subsequence at time $M(\sigma^r \oplus \sigma^{\tilde{r}})$.

Finally, the concatenation of two subsequences $\sigma^r = \langle \sigma^r_i, ..., \sigma^r_j \rangle$ and $\sigma^{\tilde{r}} = \langle \sigma^{\tilde{r}}_v, ..., \sigma^{\tilde{r}}_w \rangle$ of a second-type trip, i.e. $r, \tilde{r} = \{\pi^{\text{nd}}\}$, is also characterized by the following data:

$$DD(\sigma^r \oplus \sigma^{\tilde{r}}) = \max\{0, E(\sigma^r \oplus \sigma^{\tilde{r}}) + D(\sigma^r \oplus \sigma^{\tilde{r}}) - s_{\sigma^{\tilde{r}}_w} - l_{n+1}\} \tag{B.14}$$

Appendix C. Additional functions

The ending time $\ddot{t}^{\text{end}}$ of the first-type trip $\sigma^{\ddot{r}}$, which starts at time $\ddot{t}$ and precedes the second-type trip $\sigma^{\hat{r}}$, is calculated as in Eq. (C.1). The values $B_{\ddot{t}}(\sigma^{\hat{r}})$ and $D_{\ddot{t}}(\sigma^{\hat{r}})$ of second-type trip $\sigma^{\hat{r}}$ are calculated as in Eq. (C.2) and (C.3), respectively, with $\Delta_{\widetilde{WT}} = \max\{0, E(\sigma^{\hat{r}}) - \ddot{t}^{\text{end}}\}$ and $B_{\text{new}}(\sigma^{\hat{r}})$ calculated in complexity $\mathcal{O}(n)$.

$$\ddot{t}^{\text{end}} = \begin{cases} \ddot{t} + DM(\sigma^{\ddot{r}}) & \text{if } \ddot{t} = M(\sigma^{\ddot{r}}) \\ \ddot{t} + D(\sigma^{\ddot{r}}) & \text{otherwise} \end{cases} \tag{C.1}$$

$$B_{\ddot{t}}(\sigma^{\hat{r}}) = \begin{cases} B(\sigma^{\hat{r}}) & \text{if } \ddot{t}^{\text{end}} + \Delta_{\widetilde{WT}} \leq L(\sigma^{\hat{r}}) \\ B_{\text{new}}(\sigma^{\hat{r}}) & \text{otherwise} \end{cases} \tag{C.2}$$

$$D_{\ddot{t}}(\sigma^{\hat{r}}) = \begin{cases} D(\sigma^{\hat{r}}) & \text{if } \ddot{t}^{\text{end}} + \Delta_{\widetilde{WT}} \leq L(\sigma^{\hat{r}}) \\ D(\sigma^{\hat{r}}) - \min\{\ddot{t}^{\text{end}} + \Delta_{\widetilde{WT}} - E(\sigma^{\hat{r}}), WT(\sigma^{\hat{r}})\} & \text{otherwise} \end{cases} \tag{C.3}$$

Appendix D. Supplementary material

Supplementary material associated with this article can be found in the online version (Mendeley data).

Appendix E. Experimental design

Appendix E.1. Single-factor experiment

Five replicates of one random treatment from the 2-factor experiment are run with five possible time limits: 900, 1200, 1500, 1800, and 7200 s. Fig. E.5 presents a box plot with the results of each possible time limit in terms of the average gap with the CPLEX objective function. The x in Fig. E.5 represents the mean of the five replicates.

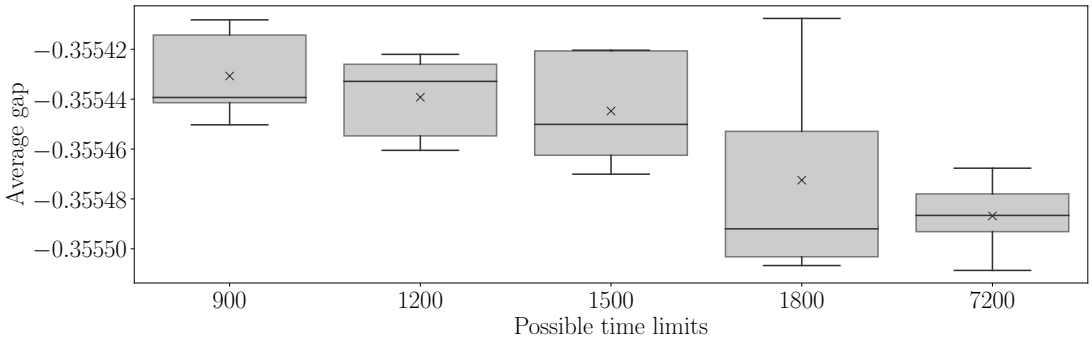

Figure E.5: Box plot with the results of the single-factor experiment

Shapiro-Wilk, Levene, and Durbin-Watson tests are applied for a significance level of 5% on the residuals of the data to verify their normality, homoscedasticity, and independence, respectively. P-values of 0.096, 0.52, and 0.21 are obtained in the tests, respectively. Then, the average gaps are normally distributed, with homogeneous variance, and independent.

The data are subjected to the ANOVA test for a significance level of 5%, which reported a p-value equals to 0.009. Therefore, it is concluded that at least one of the means of the possible time limits is different. Then, Tukey's test is applied for a significance level of 5% to find significant differences between the means of the possible time limits. Only a significant difference is found in the mean of the 7200-s and 900-s time limits with a p-value equals to 0.016. Therefore, the time limit for the 2-factor factorial experiment is set to 1200 s.

Appendix E.2. P-values of the 2-factor experiment

References

American Association of Blood Banks (2014). *Technical manual*. (18th ed.). American Association of Blood Banks.

Table E.10: P-values for the 2-factor experiment

Test	Data	Comparison	P-value	Test	Data	P-value
ANOVA	Δf_{CPLEX}	-	0.98	Shapiro-Wilk	Δf_{CPLEX} residuals	0.58
ANOVA	$\overline{iteBest}$	-	$1.74\text{x}10^{-4}$		$\overline{iteBest}$ residuals	0.29
ANOVA	$\overline{nIteNo}$	-	$3.58\text{x}10^{-5}$		$\overline{nIteNo}$ residuals	0.13
Tuckey	$\overline{iteBest}$	u-i-i:0.15 and i-i-u:0.15	$3.90\text{x}10^{-2}$	Levene	Δf_{CPLEX} residuals	0.87
		u-i-i:0.15 and i-u-i:0.15	$2.79\text{x}10^{-4}$		$\overline{iteBest}$ residuals	0.84
		i-u-i:0.25 and i-u-i:0.15	$3.53\text{x}10^{-2}$		$\overline{nIteNo}$ residuals	0.99
		i-u-i:0.2 and u-i-i:0.15	$3.31\text{x}10^{-3}$	Durbin-Watson	Δf_{CPLEX} residuals	0.39
		u-i-i:0.25 and u-i-i:0.15	$9.91\text{x}10^{-3}$		$\overline{iteBest}$ residuals	0.61
Tuckey	$\overline{nIteNo}$	u-i-i:0.15 and i-i-u:0.15	$2.05\text{x}10^{-2}$		$\overline{nIteNo}$ residuals	0.78
		u-i-i:0.15 and i-u-i:0.15	$1.04\text{x}10^{-3}$			
		i-u-i:0.2 and u-i-i:0.15	$2.61\text{x}10^{-3}$			
		u-i-i:0.25 and u-i-i:0.15	$2.53\text{x}10^{-3}$			

A significance level of 5% was considered for all tests.

American Red Cross (2017). Blood components. URL: http://www.redcrossblood.org/learn-about-blood/blood-components.

Beliën, J., & Forcé, H. (2012). Supply chain management of blood products: A literature review. *European Journal of Operational Research*, *217*, 1–16. doi:10.1016/j.ejor.2011.05.026.

Cattaruzza, D., Absi, N., & Feillet, D. (2016). Vehicle routing problems with multiple trips. *4OR*, *14*, 223–259. doi:10.1007/s10288-016-0306-2.

Chao, I.-M. M., Golden, B. L., & Wasil, E. A. (1996). The team orienteering problem. *European Journal of Operational Research*, *88*, 464–474. doi:10.1016/0377-2217(94)00289-4.

Custer, B., Johnson, E. S., Sullivan, S. D., Hazlet, T. K., Ramsey, S. D., Murphy, E. L., & Busch, M. P. (2005). Community blood supply model: Development of a new model to assess the safety, sufficiency, and cost of the blood supply. *Medical Decision Making*, *25*, 571–582. doi:10.1177/0272989X05280557.

Dantzig, G. B., & Ramser, J. H. (1959). The truck dispatching problem. *Management Science*, *6*, 80–91. doi:10.1287/mnsc.6.1.80.

Doerner, K. F., Gronalt, M., Hartl, R. F., Kiechle, G., & Reimann, M. (2008). Exact and

heuristic algorithms for the vehicle routing problem with multiple interdependent time windows. *Computers & Operations Research*, *35*, 3034–3048. doi:10.1016/j.cor.2007.02.012.

Eksioglu, B., Vural, A. V., & Reisman, A. (2009). The vehicle routing problem: A taxonomic review. *Computers & Industrial Engineering*, *57*, 1472–1483. doi:10.1016/j.cie.2009.05.009.

Fleischmann, B. (1990). The vehicle routing problem with multiple use of the vehicles, .

François, V., Arda, Y., & Crama, Y. (2019). Adaptive large neighborhood search for multitrip vehicle routing with time windows. *Transportation Science*, *53*, 1706–1730. doi:10.1287/trsc.2019.0909.

Ghandforoush, P., & Sen, T. K. (2010). A DSS to manage platelet production supply chain for regional blood centers. *Decision Support Systems*, *50*, 32–42. doi:10.1016/j.dss.2010.06.005.

Golden, B., Raghavan, S., & Wasil, E. (2008). *The vehicle routing problem: Latest advances and new challenges* volume 43. Springer US. doi:10.1007/978-0-387-77778-8.

Gunawan, A., Lau, H. C., & Vansteenwegen, P. (2016). Orienteering Problem: A survey of recent variants, solution approaches and applications. *European Journal of Operational Research*, *255*, 315–332. doi:10.1016/j.ejor.2016.04.059.

Gunpinar, S., & Centeno, G. (2016). An integer programming approach to the bloodmobile routing problem. *Transportation Research Part E: Logistics and Transportation Review*, *86*, 94–115. doi:10.1016/j.tre.2015.12.005.

Hammami, F., Rekik, M., & Coelho, L. C. (2020). A hybrid adaptive large neighborhood search heuristic for the team orienteering problem. *Computers & Operations Research*, *123*, 105034. doi:10.1016/J.COR.2020.105034.

He, M., Wei, Z., Wu, X., & Peng, Y. (2021). An adaptive variable neighborhood search ant colony algorithm for vehicle routing problem with soft time windows. *IEEE Access*, *9*, 21258–21266. doi:10.1109/ACCESS.2021.3056067.

Hernandez, F., Feillet, D., Giroudeau, R., & Naud, O. (2016). Branch-and-price algorithms for the solution of the multi-trip vehicle routing problem with time windows. *European Journal of Operational Research*, *249*, 551–559. doi:10.1016/j.ejor.2015.08.040.

Kim, H., Kim, B. I., & jin Noh, D. (2020). The multi-profit orienteering problem. *Computers & Industrial Engineering*, *149*, 106808. doi:10.1016/J.CIE.2020.106808.

Labadie, N., Prins, C., & Prodhon, C. (2016). *Metaheuristics for vehicle routing problems*. Wiley-ISTE. doi:10.1007/978-3-319-45403-0_15.

Lourenço, H. R., Martin, O. C., & Stützle, T. (2003). Iterated local search. In *Handbook of Metaheuristics* (pp. 320–353). Kluwer Academic Publishers. doi:10.1007/0-306-48056-5_11.

Mladenović, N., & Hansen, P. (1997). Variable neighborhood search. *Computers and Operations Research*, *24*, 1097–1100. doi:10.1016/S0305-0548(97)00031-2.

Mobasher, A., Ekici, A., & Özener, O. Ö. (2015). Coordinating collection and appointment scheduling operations at the blood donation sites. *Computers & Industrial Engineering*, *87*, 260–266. doi:10.1016/j.cie.2015.05.020.

Montgomery, D. C., & Runger, G. C. (2010). *Applied statistics and probability for engineers*. (Third edit ed.). John Wiley & Sons, Inc.

Murat Afsar, H., & Labadie, N. (2013). Team orienteering problem with decreasing profits. *Electronic Notes in Discrete Mathematics*, *41*, 285–293. doi:10.1016/j.endm.2013.05.104.

Neira, D. A., Aguayo, M. M., De la Fuente, R., & Klapp, M. A. (2020). New compact integer programming formulations for the multi-trip vehicle routing problem with time windows. *Computers and Industrial Engineering*, *144*. doi:10.1016/j.cie.2020.106399.

Osorio, A. F., Brailsford, S. C., & Smith, H. K. (2015). A structured review of quantitative models in the blood supply chain: A taxonomic framework for decision-making.

International Journal of Production Research, *53*, 7191–7212. doi:10.1080/00207543.2015.1005766.

Özener, O. Ö., & Ekici, A. (2018). Managing platelet supply through improved routing of blood collection vehicles. *Computers & Operations Research*, *98*, 113–126. doi:10.1016/J.COR.2018.05.011.

Pan, B., Zhang, Z., & Lim, A. (2021). Multi-trip time-dependent vehicle routing problem with time windows. *European Journal of Operational Research*, *291*, 218–231. doi:10.1016/j.ejor.2020.09.022.

Panadero, J., Juan, A. A., Bayliss, C., & Currie, C. (2020). Maximising reward from a team of surveillance drones: A simheuristic approach to the stochastic team orienteering problem. *European Journal of Industrial Engineering*, *14*, 485–516. doi:10.1504/EJIE.2020.108581.

Pirabán, A., Guerrero, W. J., & Labadie, N. (2019). Survey on blood supply chain management: Models and methods. *Computers & Operations Research*, *112*, 104756. doi:10.1016/j.cor.2019.07.014.

Prastacos, G. P. (1984). Blood inventory management: an overview of theory and practice. *Management Science*, *30*, 777–800. doi:10.1287/mnsc.30.7.777.

Rabbani, M., Aghabegloo, M., & Farrokhi-Asl, H. (2017). Solving a bi-objective mathematical programming model for bloodmobiles location routing problem. *International Journal of Industrial Engineering Computations*, *8*, 19–32. doi:10.5267/j.ijiec.2016.7.005.

Roberts, N., James, S., Delaney, M., & Fitzmaurice, C. (2019). The global need and availability of blood products: a modelling study. *The Lancet Haematology*, *6*, e606–e615. doi:10.1016/S2352-3026(19)30200-5.

Sahinyazan, F. G., Kara, B. Y., & Taner, M. R. (2015). Selective vehicle routing for a mobile blood donation system. *European Journal of Operational Research*, *245*, 22–34. doi:10.1016/j.ejor.2015.03.007.

Secretaria Distrital de Salud (2017). *Boletín estadístico anual*. Technical Report Secretaria Distrital de Salud Bogotá, Colombia.

Toth, P., & Vigo, D. (2014). *Vehicle routing: Problems, methods, and applications*. Philadelphia, PA: Society for Industrial and Applied Mathematics. doi:10.1137/1.9781611973594.fm.

Tsakirakis, E., Marinaki, M., Marinakis, Y., & Matsatsinis, N. (2019). A similarity hybrid harmony search algorithm for the Team Orienteering Problem. *Applied Soft Computing*, *80*, 776–796. doi:10.1016/J.ASOC.2019.04.038.

Vidal, T., Crainic, T. G., Gendreau, M., & Prins, C. (2013). A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time-windows. *Computers & Operations Research*, *40*, 475–489. doi:10.1016/j.cor.2012.07.018.

Vidal, T., Laporte, G., & Matl, P. (2020). A concise guide to existing and emerging vehicle routing problem variants. *European Journal of Operational Research*, *286*, 401–416. doi:10.1016/j.ejor.2019.10.010. arXiv:1906.06750.

WHO (2017). *The 2016 global status report on blood safety and availability*. Technical Report World Health Organization. doi:http://apps.who.int/iris.

Xia, Y., & Fu, Z. (2019). Improved tabu search algorithm for the open vehicle routing problem with soft time windows and satisfaction rate. *Cluster Computing*, *22*, 8725–8733. doi:10.1007/s10586-018-1957-x.

Xu, W., Liang, W., Xu, Z., Peng, J., Peng, D., Liu, T., Jia, X., & Das, S. K. (2021). Approximation Algorithms for the Generalized Team Orienteering Problem and its Applications. *IEEE/ACM Transactions on Networking*, *29*, 176–189. doi:10.1109/TNET.2020.3027434.

Yu, Q., Fang, K., Zhu, N., & Ma, S. (2019). A matheuristic approach to the orienteering problem with service time dependent profits. *European Journal of Operational Research*, *273*, 488–503. doi:10.1016/j.ejor.2018.08.007.