A mixed-integer linear programming model for the selective full-truckload multi-depot vehicle routing problem with time windows

Karim EL Bouyahyiouy[a*] and Adil Bellabdaoui[a]

---

[a]ITM – Information Technology and Management, ENSIAS - Mohammed V University in Rabat, Morocco

C H R O N I C L E	A B S T R A C T
	This article has studied a full truckload transportation problem in the context of an empty return scenario, particularly an order selection and vehicle routing problem with full truckload, multiple depots and time windows (SFTMDVRPTW). The aim is to develop a solution where a set of truck routes serves a subset of selected transportation demands from a number of full truckload orders to maximize the total profit obtained from those orders. Each truck route is a chain of selected demands to serve, originating at a departure point and terminating at an arriving point of trucks in a way that respects the constraints of availability and time windows. It is not mandatory to serve all orders, and only the profitable ones are selected. In this study, we have formulated the SFTMDVRPTW as a mixed-integer linear programming (MILP) model. Finally, Computational results are conducted on a new data set that contains thirty randomly generated problem instances ranging from 16 to 30 orders using the CPLEX software. The findings prove that our model has provided good solutions in a reasonable time.

1. Introduction

Transport has always been a pioneering sector in the economy of the market. Moreover, because of the progress of the globalization of international exchanges and the needs of carriers to meet the demands of the giant shippers, companies face everyday transport challenges with maximum efficiency of cost-competitive operations. The economic interest in the transport sector explains the interest of researchers in the optimization of routing problems. As a result of substantial collaboration between the operational research specialists (mathematical programming and combinatorial optimization) and transport managers, several implantations of optimization computer systems have been created. Optimization problems of transports are among the gorgeous achievements of operational research (OR). The problem of transport is not a recent one. Travelling salesman problem (TSP) and vehicle routing problem (VRP) are considered to have the lion's share in combinatorial optimization problems in operations research. The fundamental problem, TSP, permits to visit a set of customers with one single truck. Therefore, it plans it's tour by finding the running sequence of clients at a minimal cost. The VRP treats the case in which each customer has a specific demand and wherein several trucks can be used. As to the above problems, many constraints can be added as well to adapt to practical problems encountered in the field of transportation. Thus, every vehicle has a limited capacity (capacitated VRP, CVRP) (Lysgaard et al. 2004; Uchoa et al. 2017). The trucking company can serve its customers using a homogeneous fleet of trucks from more than one warehouse (multi-depot VRP, MDVRP) (Lahyani et al., 2019; Marín Moreno et al., 2019). These vehicles can be heterogeneous at distinct levels: the fixed vehicle utilization costs, loading capacity, variable operational costs, etc. (heterogeneous VRP, HVRP) (Bettinelli et al., 2011; Bolaños et al., 2018). Each customer must be serviced within a specified time window (VRP with time windows, VRPTW) (Solomon, 1987; Yu et al., 2011). The demand may be located on the arcs or on the nodes depending on the case (arc routing problem, ARP) (Golden et al., 1981; Ávila et al., 2017). In addition to the arc costs, each potential customer has an associated profit; the fleet of trucks and the travel time each truck can travel are limited. As a result, it is not obligatory to visit all customers, and only profitable ones are served within a certain time period (selective

---

* Corresponding author.
E-mail address: karim_elbouyahyaoui@um5.ac.ma (K. EL Bouyahyiouy)

VRP, SVRP) (Aras et al., 2011; Lahyani et al., 2017; Rincon-Garcia et al., 2017). The trucking company provides only full truckload (FTL) transportation service, where orders (transportation demands) are shipped from the location of pick-up to the location of the delivery with trucks located at one or several warehouses (full truckload VRP, FTVRP) (Ball et al., 1983; Desrosiers et al., 1988). Each time a constraint is added or subtracted, a new problem appears. The FTVRP has many real-world applications such as the transportation of materials in the public works sector (Grimault et al., 2017), collection and delivery of round timbers in the forestry industry (Gronalt & Hirsch, 2007), chemical products distribution (Ball et al. 1983), intermodal container transportation (Nossack & Pesch, 2013; Dimitriou, 2021), vaccine distribution[1], etc. Despite its prominence, FTVRP has not been studied vigorously much as other VRP variants; only a few studies worked on this problem without focusing on the empty back return of trucks. We believe that it has not been given its due merit when it comes to research in the field of OR. This is why we are motivated and interested in studying this issue of FTL.

This paper proposes a new mathematical formulation for a full truckload transportation problem in the context of an empty return scenario. The problem is selective FTVRP with time windows and multiple depots (SFTMDVRPTW). It is different from most others in the sense that only a subset of orders (transportation demands) are serviced. The objective is to come up with a solution where a set of truck routes serves a subset of selected transportation demands from a number of FTL orders while maximizing the total profit obtained from those orders. Each truck route is an arrangement of selected demands to serve, originating at a departure point and terminating at an arrival point of trucks in a way that respects the constraints of availability and time windows. In a particular context of an empty return scenario, our problem becomes apparent when a commercial truck returns from its original destination B to its original point A of departure (his home base), and after a normal delivery trip of its customers' freight has been made from A to B. An empty return is not economically viable, so the truck has to select profitable transport orders out of all the ones placed in the network. There are several concerns related to this work, such as reducing the number of empty returns, increasing efficiency in the use of equipment of transport companies, reducing the cost of operations of transport companies, reducing CO_2 emissions, etc. Noting that no matter how the constraints of the problem are, it remains NP-hard. This means that no recognized algorithm can guarantee to find, in polynomial time, the exact solution of these problems. EL Bouyahyiouy and Bellabdaoui (2016, 2017) have only presented a brief description of the problem; they have tried to adapt some heuristic approaches in a paper published in a conference proceeding that exposed the first result of a small instance, including one central depot. Thus, the contribution of this paper is as follows:

- We present a variant of the FTL transportation problem that is inspired by logistic transportation in the context of an empty return scenario, which is not treated in the literature.

- In the literature review section, we present the full truckload problems that comprise all constraints such as selective transportation demands, MD, TWs of orders, etc.

- We formulate our SFTMDVRPTW problem as a mixed-integer linear programming (MILP) model.

- We test our model on a new set of randomly generated instances using the CPLEX software. Since our problem is a new variant of full truckload transportation problem with no benchmark instances, we believe that our benchmark will be beneficial for any variant of FTVRP.

This article is organized as follows: in section 2, we describe our problem and illustrate it by means of a small example. A literature review is presented in section 3, where the contributions of this paper are also noted. Section 4 outlines the mathematical formulation for the optimization problem addressed. Section 5 proposes a set of random instances and presents numerical results obtained by the CPLEX solver on the mathematical model. The last section presents some conclusions and highlights for researchers interested in the field.

2. Description of the problem

2.1 Structure of the problem

We have described our problem on an oriented graph $G = (X, E)$, where X represents the ensemble of nodes and E is the ensemble of possible arcs representing the feasible links. The nodes are the points of extremity $\{(L_1, U_1), \ldots, (L_n, U_n)\}$ of n transportation demands (orders) that are linked to two sets of points: $D = \{D_1, \ldots, D_m\}$ and $A = \{A_1, \ldots, A_m\}$, referring to the departure and arrival locations of trucks respectively. Fig. 1. shows an illustration of our problem that includes nine orders and two trucks.
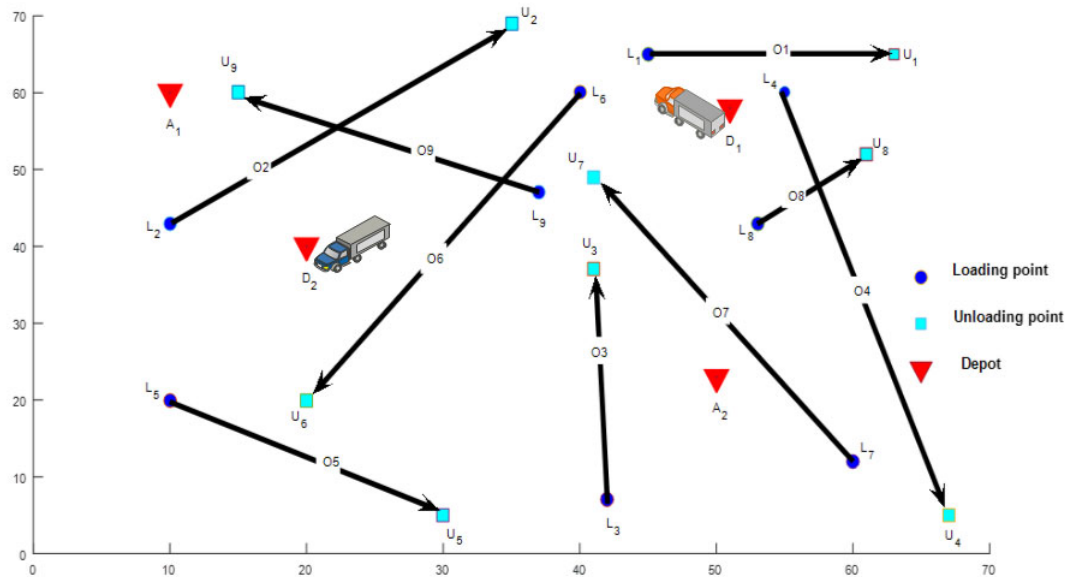
Fig. 1. An illustration of SFTMDVRPTW

2.2 The operational constraints

Selective tour. The tours may not necessarily include all orders, but rather select those that are promising to the desired objective, mainly profit.

Full truckload. An order O_i (or simply order i) consists of a full truckload. It is defined by:

- A couple of points (L_i, U_i); a loading (or pickup) point L_i and an unloading (or delivery) point U_i.

- A profit p_i.

- A distance d_i between the loading point L_i and the unloading point U_i.

Time windows. The selected orders cannot be visited at any time. This allows the client to better predict the pickup and the delivery of the order. Thus, each selected order has double time windows: $\left[L_i^{min}, \ L_i^{max}\right]$ and $\left[U_i^{min}, \ U_i^{max}\right]$. L_i^{min} and L_i^{max} are the earliest and latest time to load the goods, respectively. U_i^{min} and U_i^{max} are the earliest and latest time to perform the unloading, respectively. If a truck arrives early at a loading or unloading point, it has to wait for the start of loading or unloading time. A truck is not allowed to finish its loading (or unloading) after time L_i^{max} (or U_i^{max}). Multiple depots. Given a fixed fleet of m homogeneous trucks, each truck v_k $(k = 1, \dots, m)$ is associated with two vertices representing the points of departure and arrival of its tour, D_k and A_k respectively. Each truck is subject to the time constraints in the interval $\left[D_k^{min}, \ A_k^{max}\right]$, where D_k^{min} is the earliest departure time for truck v_k from the departure point D_k, and A_k^{max} is the latest arrival time for truck v_k at the arrival point A_k.

2.3 Objectives

The aim is to build truck routes T_k to serve a subset of selected orders from the n full truckload orders. The objective function is to maximize the net profits obtained from those orders. Each tour T_k is an arrangement of chosen orders, originating at the departure point D_k and terminating at the arrival point A_k of truck v_k so that the constraints of availability and time windows are respected.

3. Literature review

In this section, we have regrouped the diverse contributions claiming that the routing includes mainly the notion of full truckload (FTL), the selection of the truckload, and time windows constraint (TWs). This division is justified by the complexity of the problem treated and the diversity of the proposed approaches. It is also explained by the fact that only a few works have tackled the issue of the FTL. Table 1 summarizes the FTL literature and the characteristics of the present study. To our knowledge, the first efforts in the study of the FTL transportation problem are dated to the 80s. Ball et al. (1983) were the firsts that have introduced the FTVRP, which is about creating routes for private trucks and subcontracting orders of chemical products to common carriers. It can be seen as a full truckload multi-depot pickup and delivery problem

(FTMDPDP). The objective function is to reduce the cost and meet restrictions of maximum route-time. Three heuristics are suggested to resolve the problem. One of these heuristics is a greedy insertion procedure (GI); others are based on the route-first, cluster-second (RF-CS) method. Later, Desrosiers et al. (1988) presented an exact approach for the FTMDPDP, transforming it into an asymmetric traveling salesman problem (aTSP).

Table 1
Summary of the most related publications on FTVRP

Publication	Acronym	Constraints					Objective function	Solution approach		Instances	
		TW	Het	MD	S	FTL		type	Method	type	# orders
Ball et al. (1983)	FTMDPDP			●	●	●	Min. cost	H	• RF-CS • Greedy insertion	R	Up to 200
Desrosiers et al. (1988)	VRPFL			●		●	Min. EMV	E	BB	RG	Up to 104
Wang and Regan (2002)	FLPDPTW	●		●	●	●	Min. EMV	H	WPB	R	Up to 75
Arunapuram et al. (2003)	VRPFL	●		●		●	Min. cost	E	CG	RG	Up to 200
Gronalt et al. (2003)	FLPDPTW	●		●		●	Min. EMV	H	savings algorithm	RG	Up to 512
Currie and Salhi (2003)	FTPDPTW	●	●	●		●	Min. cost	E H	• MILP • 3-phase CH	R RG	Up to 208 Up to 500
Currie and Salhi (2004)	FTPDPTW	●	●	●		●	Min. cost	SSBM	TS	R RG	Up to 208 Up to 500
Jula et al. (2005)	FTPDPTW	●				●	Min. cost	E PBM	• 2Phase DP • Hybrid DP with AG	RG	Up to 20 Up to 100
Imai et al. (2007)	VRPFC		●			●	Min. cost	H	LG	RG	Up to 200
Gronalt and Hirsch (2007)	FT-PDPTW	●		●		●	Min. EMV	SSBM	TS	R	Up to 85
Caris and Janssens (2009)	FTPDPTW	●				●	Min. cost	H	• 2PIH • LS	RG	Up to 200
Zhang et al. (2009)	am-TSPTW	●		●		●	Min. RD	E H SSBM	• CPLEX • Cluster • RTS	RG	Up to 200
Liu et al. (2010b)	SFTVRP				●	●	Min. cost	PBM	MA	RG	Up to 400
Zhang et al. (2010)	am-TSPTW	●		●		●	Min. RD	H SSBM	WPB RTS	RG	Up to 75
Liu et al. (2010a)	FTMDCVRP			●		●	Min. EVM	H	2PH	RG	Up to 300
Venkateshan and Mathur (2011)	FTPDP		●	●		●	Min. cost	E	CG	RG	Up to 20
Braekers et al. (2013)	FTPDPTW	●				●	Min. NV Min. TD	SSBM	DA	RG	Up to 200
Nossack and Pesch (2013)	FTPDPTW	●		●		●	Min. RD	H	2-stage heuristic	RG	Up to 75
Braekers et al. (2014)	FTPDPTW	●				●	Min NV Min. TD	H SSBM SSBM	• IM • DA • hybrid DA with TS	RG	Up to 200
Grimault et al. (2017)	FTPDP-RS	●	●	●		●	Min. cost	SSBM	ALNS	R	Up to 85
Xue et al. (2021)	MS- FTPDPTW	●				●	Min. cost	SSBM	hybrid CG with GA/VNS	RG	Up to 127
This study	SFTMDVRPTW	●		●	●	●	Max. profit	E	CPLEX	RG	Up to 30

Het: heterogeneous fleet, MD: multi-depot, S: selective, RD: route duration, TD: total distance, NV: number of vehicles, EVM: Empty-Vehicle-Movement, E: exact, H: heuristic, SSBM: single solution based metaheuristic, PBM: population-based metaheuristic, DA: deterministic annealing algorithm, R: real data, RG: randomly generated data.

There are no TWs requirements at pickup and delivery points, but a time limit is imposed on truck routes. The goal is to minimize empty movements of trucks. To resolve the problem, a branch-and-bound (BB) method is applied. Arunapuram et al. (2003) have introduced FTVRP with multiple depots and pickup TWs. They formulated the problem as an integer programming (IP) formulation, and they developed a column generation (CG) approach inside a BB method to solve it.

Gronalt et al. (2003) have presented an exact formulation for FTPDPTW where full truckload orders are moved from and to distribution centers. Trucks are available at various warehouses and can be assigned several tours throughout the time of planning. The authors developed four different savings based heuristics that are compared on randomly generated instances. Venkateshan and Mathur (2011) have extended the FTVRP for heterogeneous trucks, where multiple visits are required by the same or different trucks to satisfy each order. Relying on the notations of Arunapuram et al. (2003), Venkateshan and Mathur (2011) formulated this problem as an IP problem, and they developed an exact approach from CG to resolve it. Currie and Salhi (2003, 2004) have tackled FTPDPTW with heterogeneous goods and trucks, where the location of orders' pickup is undetermined. Their 2003 work presented a MILP model whose objective is to minimize the total costs. Then, they proposed a hybrid method combining a GI heuristic with three neighborhood operators. Thereafter, these three operators would be used in a tabu search (TS) algorithm (Currie and Salhi 2004). Gronalt and Hirsch (2007) studied a FTMDPDPTW in the context of log-truck scheduling with the goal of minimizing the total duration of trucks' empty movements. They have proposed a TS to solve 20 instances with 30 to 85 orders and 9 to 28 trucks. Liu et al. (2010a) proposed an exact formulation and a two-phase GI heuristic to study FTL capacitated ARP with multiple depots (MDCARPFL) in the context of carrier collaboration with the aim of reducing empty movements of trucks. Liu et al. (2010b) developed a memetic algorithm for FVRP with order selection in collaborative transportation. It is assumed that carriers receive two types of orders: one given by their customers, while the other by outer carriers. The orders from the first type have to be served by one of the carriers' internal vehicles or assigned to an external collaborative transporter, but a penalty cost is incurred. For each order given by outer partners, the carriers can refuse or process it with a compensative payment. The goal is to come up with a set of private truck routes that serve a subset of selected orders, aiming at minimizing the total cost minus the compensative payments for accepting orders from other carriers. Grimault et al. (2017) investigated the FTPDPTW with resource synchronization that appears in the context of public-work companies. The problem consists of optimizing the transport of materials between sites, using a fleet of heterogeneous trucks. The objective is to minimize the total cost (traveling, service time, truck utilization). They propose an Adaptive Large Neighborhood Search (ALNS) to solve this problem. The solving method ALNS was tested on real instances.

Container drainage problem (CDP) represents a special class of FTPDP (Braekers et al. 2013). The goal is to carry containers, representing full truckloads, between customers and container terminals. When there are TWs at customers or container terminals, it becomes FTPDPTW. When we load a merchandise at a loading point in the FTL case, we are obliged to unload it at the unloading point in the next step. So, the FTPDPTW can be transformed into an asymmetric multiple traveling salesman problem with time windows (am-TSPTW) by considering each pickup-delivery pair as a node (Wang & Regan 2002; Jula et al. 2005; Zhang et al. 2009; Braekers et al. 2013). Wang and Regan (2002) studied FTPDPTW, in which only pickup TWs are considered. The objective is to reduce the empty truck movement cost while the number of served orders within their TWs is as large as possible. The authors developed an iterative method for solving the problem using the window-partition-based (WPB) method. Later, Jula et al. (2005) extended the earlier work to formulate the container movement by trucks with TWs at both pickup and delivery locations. The authors propose and compare three solving approaches: an exact two-phase algorithm based on dynamic programming (DP) for small instances with up to 20 orders, a hybrid approach consisting of DP and GA, and an insertion heuristic method for large problem instances with up to 100 orders. Imai et al. (2007) addressed the same problem of Gronalt et al. (2003) without TWs, arising during the pickup and delivery of full containers that transport them between intermodal terminals. They present a new formulation for this problem as a 0-1 linear programming model (LP) and propose a Lagrangian relaxation (LR) to solve the mathematical formulation. Later, Caris and Janssens (2009) have extended the problem definition of Imai et al. (2007) to FTPDPTW by imposing hard TWs at the customer locations and the depot. Their solution approach employed a two-phase insertion heuristic to construct an initial solution, which was then improved by a local search. Zhang et al. (2009) defined an integrated CDP that handles full and empty containers between customers, one single terminal and multiple depots. Trucks don't have to return to their departure depot; hard TWs are defined at the customer locations and the terminal. The objective is to minimize total travel time. The problem is formulated as am-TSPTW with multiple depots and is solved by a clustering method and a reactive tabu search (RTS) algorithm. Zhang et al. (2010) extended the work of Zhang et al. (2009) to multiple terminals and multiple depots. Wang and Regan (2002) presented a modified WPB method as a solution approach for the problem. The proposed method has been tested on randomly generated instances with up to 75 orders and compared with the RTS in Zhang et al. (2009). Inspired by Zhang et al. (2010), Nossack and Pesch (2013) came with a new formulation for FTPDPTW that aims at minimizing the total operating time of all trucks in use. They propose a two-phase heuristic solution approach for solving the FTPDPTW. The results indicate that their 2-stage heuristic outperforms the WPB method used by Zhang et al. (2010) in terms of CPU time and solution quality. Braekers et al. (2013, 2014) studied FTVRP in a drainage operation that is very similar to the ones reviewed by Zhang et al. (2009, 2010). Braekers et al. (2013) propose two formulations that are based on an am-TSPTW problem. A hierarchical objective function, which first minimizes the number of vehicles and second minimizes the total distance, is proposed. For both formulations, they develop a single-phase and a two-phase DA. Braekers et al. (2014) extended their previous work by considering, for the first time, two objectives with equal priority: minimizing the number of vehicles and minimizing total distances. They propose three solution algorithms: iterative method, DA, and hybrid of DA and TS. They concluded that among the three methods, the hybrid algorithm DA-TS yielded the best results. Xue et al. (2021) presented a hybrid CG with GA and VNS for the FTVRPTW with multiple shifts; TWs of transportation containers cover several shifts, and each order's shift identification is part of decision-making.

This article describes a mathematical formulation for the FTL transportation problem, particularly an order selection and vehicle routing problem with full truckload, multiple depots and time windows (SFTMDVRPTW). The problem is characterized by several attributes: FTL, selective orders, loading and unloading TWs of the orders, waiting times, profit maximization, departure and arrival points for each truck subjected to time limit constraints. In contrast to most of the existing studies on FTL, the departure time of each truck from its departure point is a decision variable. The selective aspect, which relaxes the constraint that all orders should be serviced, refers to the situation where commercial trucks return back empty and must select certain orders out of all the orders placed in the network. However, Ball et al.'s, Wang and Regan's and Liu et al.'s studies deal with the selective orders in the context of carrier collaboration, mainly the exchange of orders between carriers. There are three highlights in our work. First, we introduce an original selective FTL transportation problem, which combines several attributes as shown in Table 1, and accounts for characteristics that are not yet addressed in the literature but required in actual applications. Second, a new MILP model is formulated for this problem. Third, CPLEX was used to solve the MILP model on a new set of test instances.

4. Mathematical formulation

To formulate the proposed MILP model, parameters, and decision variables are presented below.

Parameters
m	Number of trucks
v_k	Truck of index k; $k = 1, \dots, m$
D_k	Departure point of truck k
D_k^{min}	Earliest departure time of truck k
A_k	Arrival point of truck k
A_k^{max}	Latest arrival time of truck k
n	Number of orders
O_i	Order of index i; $i = 1, \dots, n$
p_i	Revenue associated with order i
(L_i, U_i)	Couple of points of loading and unloading of the order i
$[L_i^{min}, L_i^{max}]$	Loading time window of the order i
$[U_i^{min}, U_i^{max}]$	Unloading time window of the order i
d_i	Distance between loading and unloading point of order i
c_i	Loaded traveling cost to serve the order O_i
c^a	Wait cost per unit time before an order's loading or unloading
c_{ij}	Cost of empty travel between delivery location of order O_i and pickup location of order O_j
$c_{0,i}^k$	Cost of travel from start depot D_k to loading location L_i of order O_i
$c_{i,n+1}^k$	Cost of travel from delivery location of order i to arriving depot A_k
t_i	Traveling time between loading and unloading points of order i
t_{ij}	Traveling time from the delivery location of the order O_i to the pickup location of the order O_j
$t_{0,i}^k$	Traveling time from the start depot D_k to pickup location L_i of the order i
$t_{i,n+1}^k$	Traveling time from the delivery location of order i to arriving location A_k
M	A big number

Decision variables
x_{ij}^k	Boolean variable $x_{ij}^k = 1$ if truck k travels from the unloading point of order O_i to loading point of O_j order
x_{0i}^k	Boolean variable $x_{0i}^k = 1$ If truck k serves order O_i as its first order
$x_{i,n+1}^k$	Boolean variable $x_{i,n+1}^k = 1$ If truck k serves order O_i as its final order
$x_{0,n+1}^k$	Boolean variable $x_{0,n+1}^k = 1$ If no orders assigned to truck k
$t_{i,L}^k$	Start time to load order i on the truck k
$t_{i,U}^k$	Start time to unload order i of truck k
$a_{i,L}^k$	Waiting time before loading order i on truck k
$t_{0,L}^k$	Departure time of truck k from D_k
$t_{n+1,U}^k$	Arriving time of truck k at A_k

The MILP model of the SFTMDVRPTW problem is given as follows:

$$\max \sum_{k=1}^{m}\sum_{i=1}^{n}\sum_{j=1}^{n+1} p_i x_{ij}^k - \sum_{k=1}^{m}\sum_{i=1}^{n}\sum_{j=1}^{n+1} c_i x_{ij}^k$$

$$-\sum_{k=1}^{m}\sum_{j=1}^{n+1} c_{0,j}^k x_{0,j}^k - \sum_{k=1}^{m}\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}^k x_{ij}^k - \sum_{k=1}^{m}\sum_{i=1}^{n} c_{i,n+1}^k x_{i,n+1}^k - c^a \sum_{k=1}^{m}\sum_{i=1}^{n} (a_{i,L}^k + t_{i,U}^k - t_{i,L}^k - t_i)$$

(1)

subject to

$$D_k^{\min} \le t_{0,L}^k \qquad\qquad \forall k = 1,...,m \tag{2}$$

$$t_{n+1,U}^k \le A_k^{\max} \qquad\qquad \forall k = 1,...,m \tag{3}$$

$$L_i^{\min} \le t_{i,L}^k \le L_i^{\max} \qquad \forall i = 1,...,n, \ \ \forall k = 1,...,m \tag{4}$$

$$U_i^{\min} * \sum_{j=1}^{n+1} x_{ij}^k \le t_{i,U}^k \le U_i^{\max} \qquad \forall i = 1,...,n, \ \ \forall k = 1,...,m \tag{5}$$

$$t_{i,L}^k + t_i \le t_{i,U}^k \qquad\qquad \forall i = 1,...,n, \ \ \forall k = 1,...,m \tag{6}$$

$$t_{0,L}^k + t_{0,i}^k + a_{i,L}^k \le t_{i,L}^k + M * (1 - x_{0i}^k) \qquad \forall i = 1,...,n, \ \ \forall k = 1,...,m \tag{7}$$

$$t_{0,L}^k + t_{0,i}^k + a_{i,L}^k \ge t_{i,L}^k - M * (1 - x_{0i}^k) \qquad \forall i = 1,...,n, \ \ \forall k = 1,...,m \tag{8}$$

$$t_{i,U}^k + t_{i,j}^k + a_{j,L}^k \le t_{j,L}^k + M * (1 - x_{ij}^k) \qquad \forall i,j = 1,...,n, \ \ \forall k = 1,...,m \tag{9}$$

$$t_{i,U}^k + t_{i,j}^k + a_{j,L}^k \ge t_{j,L}^k - M * (1 - x_{ij}^k) \qquad \forall i,j = 1,...,n, \ \ \forall k = 1,...,m \tag{10}$$

$$t_{i,U}^k + t_{i,n+1}^k \le t_{n+1,U}^k + M * (1 - x_{i,n+1}^k) \qquad \forall i = 1,...,n, \ \ \forall k = 1,...,m \tag{11}$$

$$t_{i,U}^k + t_{i,n+1}^k \ge t_{n+1,U}^k - M * (1 - x_{i,n+1}^k) \qquad \forall i = 1,...,n, \ \ \forall k = 1,...,m \tag{12}$$

$$\sum_{j=1}^{n+1}\sum_{k=1}^{m} x_{ij}^k \le 1 \qquad\qquad \forall i = 1,...,n \tag{13}$$

$$\sum_{i=0}^{n}\sum_{k=1}^{m} x_{ij}^k \le 1 \qquad\qquad \forall j = 1,...,n \tag{14}$$

$$\sum_{j=1}^{n+1} x_{0j}^k = 1 \qquad\qquad \forall k = 1,...,m \tag{15}$$

$$\sum_{i=0}^{n} x_{ih}^k - \sum_{j=1}^{n+1} x_{hj}^k = 0 \qquad \forall h = 1,...,n, \ \ \forall k = 1,...,m \tag{16}$$

$$\sum_{i=0}^{n} x_{i,n+1}^k = 1 \qquad\qquad \forall k = 1,...,m \tag{17}$$

$$x_{i,0}^k = 0 \qquad\qquad \forall k = 1,..,m, \ \ \forall i = 0,...,n+1 \tag{18}$$

$$x_{n+1,i}^k = 0 \qquad\qquad \forall k = 1,..,m, \ \ \forall i = 0,...,n+1 \tag{19}$$

$$x_{i,i}^k = 0 \qquad\qquad \forall k = 1,..,m, \ \ \forall i = 0,...,n+1 \tag{20}$$

$$x_{i,j}^k + x_{j,i}^k \le 1 \qquad \forall i,j = 0,...,n+1, \quad \forall k = 1,..,m, \tag{21}$$

$$x_{ij}^k \in \{0,1\} \qquad \forall i,j = 0,...,n+1, \ \forall k = 1,...,m \tag{22}$$

$$t_{i,L}^k \ge 0 \qquad\qquad \forall i = 0,...,n, \ \ \forall k = 1,...,m \tag{23}$$

$$t_{i,U}^k \ge 0 \qquad\qquad \forall i = 1,...,n+1, \ \ \forall k = 1,...,m \tag{24}$$

$$a_{i,L}^k \ge 0 \qquad\qquad \forall i = 1,...,n, \ \ \forall k = 1,...,m \tag{25}$$

The objective function (1) maximizes the total profit, which is equal to total revenue from the selected orders (first term) minus the overall transportation cost of the routes. The transportation costs are composed of the cost of moving loaded trucks (second term), empty cost (third, fourth and fifth terms), and cost of waiting (sixth term). Constraints (2)-(5) ensure the feasibility of the time schedule; each truck route has to meet the time windows at all loading and unloading locations. Constraints (6) impose that the time of unloading has to be more than the summation of the loading time and the service time of the order. This also implies that the waiting time $t_{i,U}^{k} - t_{i,L}^{k} - t_i$ of truck k before unloading the order i is a positive integer. Constraints (7) and (8) require that loading of the order O_i on truck k may not start before moving the truck to the start depot D_k. Constraints (9) and (10) stipulate that if $x_{ij}^{k} = 1$, then $t_{j,L}^{k} = t_{i,U}^{k} + t_{i,j} + a_{j,L}^{k}$. This means that the loading of the order j can't start unless the order i is serviced and the displacement has happened. Constraints (11) and (12) require that the vehicle can transport the demand if it is likely to reach the arriving depot before the latest time. Notice that the decision variables $t_{i,L}^{k}$ and $t_{i,U}^{k}$ increase along the route. Therefore, constraints (6) - (12) can also eliminate sub-tours among order points, are similar to the Miller–Tucker–Zemlin's constraints (Desrochers and Laporte 1991) prohibit trucks from returning to order O_i if already served on the same tour. Constraints (13) and (14) necessitate that each time a vehicle enters the pickup or the delivery location, only one exit to any other location (pickup, delivery or arriving locations) can be performed. Constraints (15)-(17) specify the routing constraints. Each vehicle could depart from one starting depot (constraint 15). Once a truck loads a merchandise, it must reach the unloading location in the next step (constraint 16). Each vehicle has to go to the arriving depot (constraint 17). Constraints (18) and (19) entail that no vehicle can return to its start depot, and the arriving depot is the last point of each truck route. Constraints (20) eliminate cycles at the orders. Constraints (21) states that either O_i precedes O_j or vice versa or orders i and j are not into the same tour. Constraints (22)-(25) declare the decision variables.

5. Experimental tests

This section introduces and discusses the computational findings of our mathematical model on newly generated test instances. First, we describe the new problem instances used to test our MILP model. Afterwards, we illustrate the computational findings in the example designed in section 2 and on the generated instances.

5.1. Instances generation

Since the SFTMDVRPTW is a new problem, there are no benchmark instances available to the authors' knowledge. It is challenging to have access to authentic data related to the transportation sector. Therefore, we have generated some instances based on Solomon's VRPTW benchmark datasets[2]. The latter is divided into three instance categories: R, C, RC. Each instance comprises 100 points, in which the coordinates of each point are distributed over a $[0, 100]^2$ square. Each category is defined according to how the points are located within the square (R for remote, C for cluster and RC for a mix of remotely and clustered). To generate small and medium instances in analogy to real-life data, we have considered the first 25 and 50 points of the classes C, R, and RC. First, we generate the coordinates of all points. Thereafter, to generate a set of orders, we randomly decide whether a point is considered a loading or unloading point. To generate the departure and arrival points of trucks, we divide the $[0, 100]^2$ square into four quarters. We randomly located the departure point in a quarter and the arrival point in another one so that the distance between them is greater than 60. The traveling distance and the traveling cost (loaded or empty) are calculated using the Euclidean distance. The unit cost of waiting is set to $c^a = 0.42$. The duration t_i required for servicing the order O_i is equal to the distance between the departure and the arrival locations of the order i plus loading and unloading time of order O_i. The profit p_i of each order O_i is set to $p_i = 6 \times d_i$. The width of each time window, denoted by WTW, ranges from 2 to 4h. The loading time window $\left[L_i^{min}, L_i^{max} \right]$ of each order O_i is set as:

$$L_i^{min} = \min_{k=1,\dots,m} \{d(D_k, L_i)\} + \left\lfloor \left(A_k^{max} - \min_{k=1,\dots,m} \{d(D_k, L_i)\} - t_i - \max_{k=1,\dots,m} \{d(U_i, A_k)\} \right) \times U(0,1) \right\rfloor \quad \text{and}$$

$L_i^{max} = L_i^{min} + WTW$. The unloading time window $\left[U_i^{min}, U_i^{max} \right]$ is set as $U_i^{min} = L_i^{min} + t_i$ and $U_i^{max} = U_i^{min} + WTW$.

Table 2 shows the main characteristics for each generated instance; it specifies the number of first points considered from the original instances of Solomon, number of orders, number of trucks, service time (loading/unloading time), earliest departure and latest arrival times permitted for each truck, and the WTW.

---

[2] website: http://web.cba.neu.edu/~msolomon/problems.htm

Table 2
Main characteristics of the generated instances

Instance	# of points	# of orders	# of trucks	Service time (in minutes)	D_k^{min} (in minutes)	A_k^{max} (in minutes)	WTW (in minutes)
SFTL1/C25/16/2	25	16	2	40	0	720	240
SFTL1/C50/24/3	50	24	3	40	0	720	240
SFTL2/C25/16/2	25	16	2	40	0	720	240
SFTL2/C50/24/3	50	24	3	40	0	720	240
SFTL3/C25/16/2	25	16	2	40	0	720	240
SFTL3/C50/24/3	50	24	3	40	0	720	240
SFTL4/C25/16/2	25	16	2	30	0	600	180
SFTL4/C50/24/3	50	24	3	30	0	600	180
SFTL5/C25/16/2	25	16	2	30	0	600	180
SFTL5/C50/24/3	50	24	3	30	0	600	180
SFTL1/R25/20/2	25	20	2	10	0	480	120
SFTL1/R50/30/3	50	30	3	10	0	480	120
SFTL2/R25/20/2	25	20	2	10	0	480	120
SFTL2/R50/30/3	50	30	3	10	0	480	120
SFTL3/R25/20/2	25	20	2	20	0	720	180
SFTL3/R50/30/3	50	30	3	20	0	720	180
SFTL4/R25/20/2	25	20	2	20	0	720	180
SFTL4/R50/30/3	50	30	3	20	0	720	180
SFTL5/R25/20/2	25	20	2	20	0	720	180
SFTL5/R50/30/3	50	30	3	20	0	720	180
SFTL1/RC25/20/2	25	20	2	10	0	480	120
SFTL1/RC50/30/3	50	30	3	10	0	480	120
SFTL2/RC25/20/2	25	20	2	10	0	480	120
SFTL2/RC50/30/3	50	30	3	10	0	480	120
SFTL3/RC25/20/2	25	20	2	20	0	720	180
SFTL3/RC50/30/3	50	30	3	20	0	720	180
SFTL4/RC25/20/2	25	20	2	20	0	720	180
SFTL4/RC50/30/3	50	30	3	20	0	720	180
SFTL5/RC25/20/2	25	20	2	20	0	720	180
SFTL5/RC50/30/3	50	30	3	20	0	720	180

We set the format of each instance as follows: $SFTLx/Source/n/m$. Consider the instance $SFTL1/C25/16/2$ as an example. There are 16 orders and 2 trucks. The code $C25$ means this instance is constructed by using only the 25 first points of the original Solomon's instance category C, and the code $SFTL1$ is the instance's number. Ten instances are generated within each type of instance category, while the number of orders ranges from 16 to 30 and the number of trucks ranges from two to three. These newly generated test instances can be downloaded via link[3].

5.2. Computational results

The MILP model of Section 4 was implemented in the OPL and solved using IBM ILOG CPLEX 12.5. For each instance, we run the CPLEX with its default parameter settings until finding an optimal solution, or until exhausting the memory. All experiments were performed on a PC with Intel(R) Core(TM) i5 Processor (2.4 GHz) and 4 GB RAM operating the Windows 10 Professional with 64 bits.

5.2.1. Result on the illustrative example

The illustrative example presented in Fig. 1 is used to validate the proposed MILP. There are 22 points where the coordinates of each point are given in Table 3. The first 18 points are randomly taken from Solomon's instance $R101$ and randomly paired off to form nine orders. Loading points L_i are represented by circles and unloading points U_i by squares. The last four points represent the points of departure and arrival of the two trucks. Data of the two trucks, loading and unloading points and loading and unloading time windows of the nine orders are given in Table 4.

Table 3
Coordinates of points

Point	1	2	3	4	5	6	7	8	9	10	11
Coordinates	(41, 49)	(10, 43)	(55, 60)	(30, 5)	(10, 20)	(15, 60)	(45, 65)	(41, 37)	(40, 60)	(35, 69)	(63, 65)
Point	12	13	14	15	16	17	18	19	20	21	22
Coordinates	(20, 20)	(60, 12)	(42, 7)	(67, 5)	(37, 47)	(53, 43)	(61, 52)	(51, 58)	(20, 40)	(50, 23)	(10, 60)

---

[3] (https://drive.google.com/file/d/1SR5Jk3G4hwwlpe-SCeZwetKmF0558PnJ/view?usp=sharing)

Table 4
Pickup and delivery points, time windows of the orders (WTW=60 minutes), departure and arrival points, earliest departure and latest arrival times of the two trucks

Order O_i	1	2	3	4	5	6	7	8	9	Truck 1	Truck 2
L_i	7	2	14	3	5	9	13	17	16	-	-
U_i	11	10	8	15	4	12	1	18	6	-	-
L^i_{min}	60	50	250	115	180	100	215	150	340	-	-
L^i_{max}	120	110	310	175	240	160	275	210	400	-	-
U^i_{min}	98	106	300	191	225	165	277	182	386	-	-
U^i_{max}	158	166	360	251	285	225	337	242	446	-	-
D_k	-	-	-	-	-	-	-	-	-	19	20
A_k	-	-	-	-	-	-	-	-	-	22	21
D^{min}_k	-	-	-	-	-	-	-	-	-	0	0
A^{max}_k	-	-	-	-	-	-	-	-	-	480	480

The results obtained with the solver CPLEX on the illustrative example are shown in Fig. 2 and in Table 5. The truck 1 served the transportation demands 1,4, 7 and 9 respectively. Truck 2 has transported the orders 2, 6, 5 and 3; the order 8 is unselected. The realized profit is raised to 1359.96 and the total computation time is 1.62 seconds.
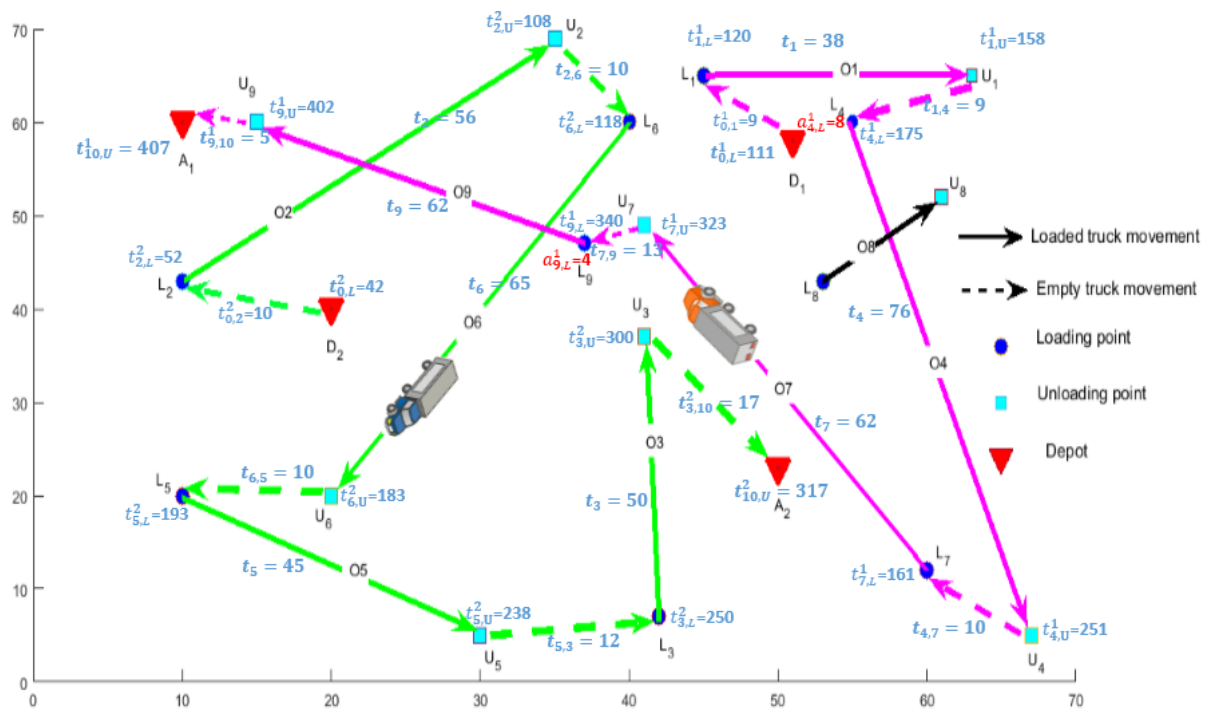

Fig. 2. Result on the illustrative example of SFTMDVRPTWP

Table 5
Result on the illustrative example

Route	Time of departure	Time of arrival	Time duration			Cost			Revenue	Profit per route
			Travel		Waiting					
			Loaded	Empty		Loaded	Empty	Waiting		
v_1: D_1-O_1-O_4-O_7-O_9-A_1	111	407	238	46	12	158	46	5.04	948	738.96
v_2: D_2-O_2-O_6-O_5-O_3-A_2	42	317	216	59	0	136	59	0	816	621
Total	-	-	454	105	12	294	105	5.04	1764	1359.96

### 5.2.2. Results and discussion for the generated instances

Table 6 summarizes the results performed on the 30 generated instances. Column one presents the instances' name. The second, third and fourth ones indicate respectively the binary number, continuous variables and constraints for each instance. The fifth and sixth columns are the upper bounds (UB_{MILP}) and the best integer solutions (LB_{MILP}) obtained by CPLEX, respectively. The seventh column represents the gap between UB_{MILP} and LB_{MILP} provided by CPLEX. The eighth column is the CPU-times of CPLEX. The ninth column is the number of nodes explored in the search tree. The tenth column shows
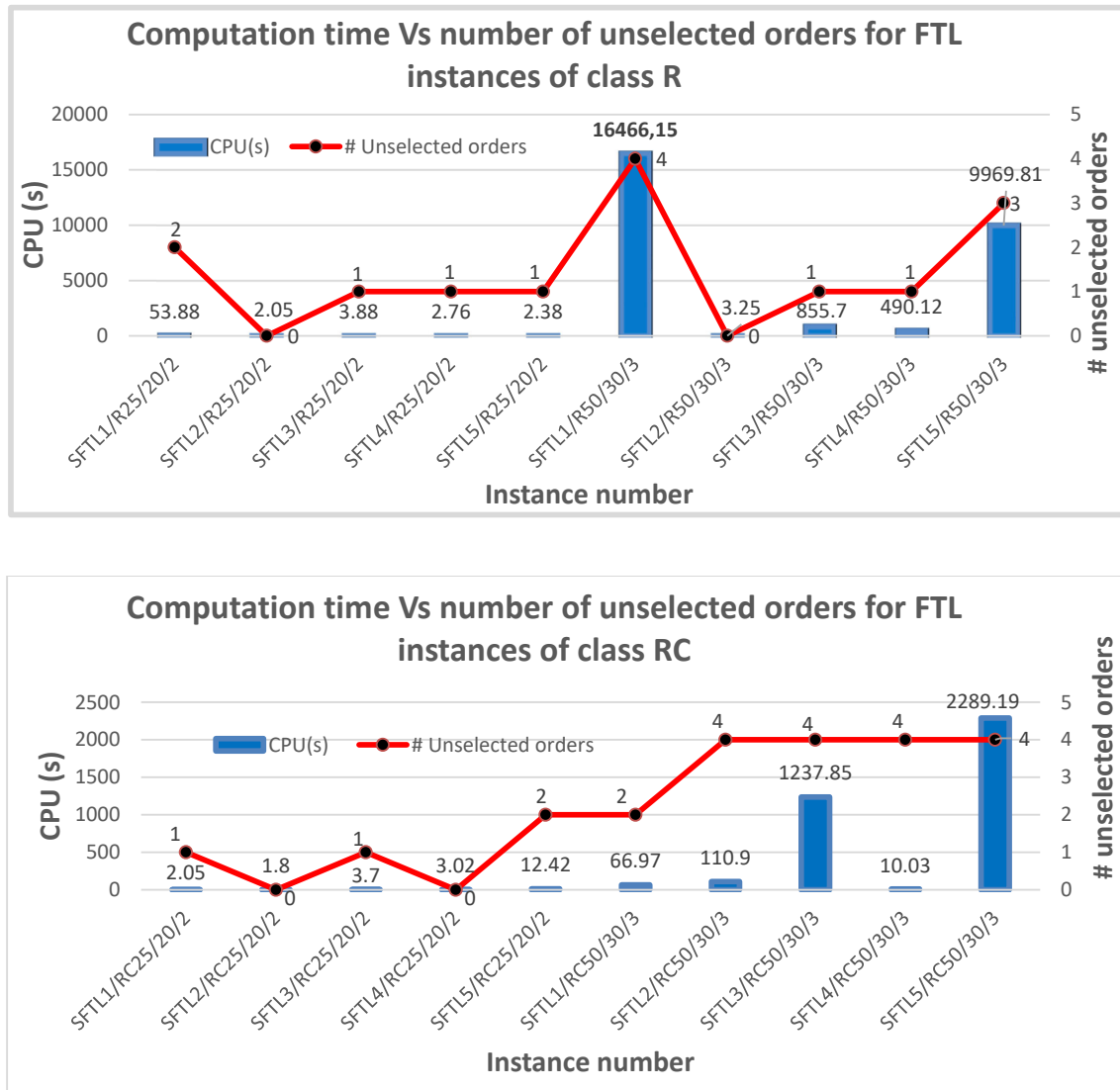
the number of orders not selected in the optimal solutions found by CPLEX. As observed in Table 6, the gap between UB_{MILP} and LB_{MILP} for all instances, except $SFTL1/R50/30/3$ instance, equals zero. This means that the MILP model found optimal solutions. For the $SFTL1/R50/30/3$ instance, the gap equals 5.58 percent. The average running time of CPLEX for all instances is about 330 seconds. Most of the instances (23 out of 30) can be solved in less than 3 minutes. Fig. 3 illustrates a comparison between CPU time and the number of unselected orders in the optimal solution from CPLEX for test instances. For the FTL instances of class C, when $WTW = 240$ and $A_k^{max} = 720$, for $n = 16$ and $m = 2$, CPU times to solve the instances $SFTL1 - 3/C25/16/2$ are nearly the same, which doesn't exceed 6 seconds. Also, the exact number of unselected orders, which is 2, is recorded as found by the optimal solution. The same can be said when

Table 6
Computational results for 30 generated instances

| Instance | variables | | # Constr. | LB_{MILP} | UB_{MILP} | Gap (%) | CPU(s) | # Nodes | # Unselected orders |
	# Bin.	# Cont.							
$SFTL1/C25/16/2$	648	101	2108	819	819*	0.00	5.44	4515	2
$SFTL1/C50/24/3$	2028	223	6426	1067	1067*	0.00	66.08	23542	3
$SFTL2/C25/16/2$	648	101	2108	766	766*	0.00	4.34	2779	2
$SFTL2/C50/24/3$	2028	223	6426	1215	1215*	0.00	14.07	4771	3
$SFTL3/C25/16/2$	648	101	2108	828	828*	0.00	4.01	2987	2
$SFTL3/C50/24/3$	2028	223	6426	1375	1375*	0.00	66.18	46117	3
$SFTL4/C25/16/2$	648	101	2108	520	520*	0.00	23.88	20433	1
$SFTL4/C50/24/3$	2028	223	6426	894	894*	0.00	4.72	1114	1
$SFTL5/C25/16/2$	648	101	2108	810	810*	0.00	2.29	1134	1
$SFTL5/C50/24/3$	2028	223	6426	1100	1100*	0.00	717.92	130621	5
$SFTL1/R25/20/2$	968	125	3108	2130	2130*	0.00	53.88	26188	2
$SFTL1/R50/30/3$	3072	277	9642	(3120)	(3294.04)	(5.58)	(16466.15)	(2967072)	(4)
$SFTL2/R25/20/2$	968	125	3108	1346	1346*	0.00	2.05	0	0
$SFTL2/R50/30/3$	3072	277	9642	2346	2346*	0.00	3.25	174	0
$SFTL3/R25/20/2$	968	125	3108	2331	2331*	0.00	3.88	2424	1
$SFTL3/R50/30/3$	3072	277	9642	3494	3494*	0.00	855.70	233772	1
$SFTL4/R25/20/2$	968	125	3108	2176	2176*	0.00	2.76	821	1
$SFTL4/R50/30/3$	3072	277	9642	3398	3398*	0.00	490.12	83742	1
$SFTL5/R25/20/2$	968	125	3108	2488.2	2488.2*	0.00	2.38	201	1
$SFTL5/R50/30/3$	3072	277	9642	3742	3742*	0.00	9969.81	1381457	3
$SFTL1/RC25/20/2$	968	125	3108	1857	1857*	0.00	2.05	0	1
$SFTL1/RC50/30/3$	3072	277	9642	2882	2882*	0.00	66.97	28488	2
$SFTL2/RC25/20/2$	968	125	3108	1538	1538*	0.00	1.80	0	0
$SFTL2/RC50/30/3$	3072	277	9642	2889	2889*	0.00	110.90	20437	4
$SFTL3/RC25/20/2$	968	125	3108	2246	2246*	0.00	3.70	1870	1
$SFTL3/RC50/30/3$	3072	277	9642	4193	4193*	0.00	1237.85	130732	4
$SFTL4/RC25/20/2$	968	125	3108	1859	1859*	0.00	3.02	170	0
$SFTL4/RC50/30/3$	3072	277	9642	3852	3852*	0.00	10.03	2712	4
$SFTL5/RC25/20/2$	968	125	3108	2458	2458*	0.00	12.42	5877	2
$SFTL5/RC50/30/3$	3072	277	9642	4056	4056*	0.00	2289.19	298813	4

* indicate optimal solution
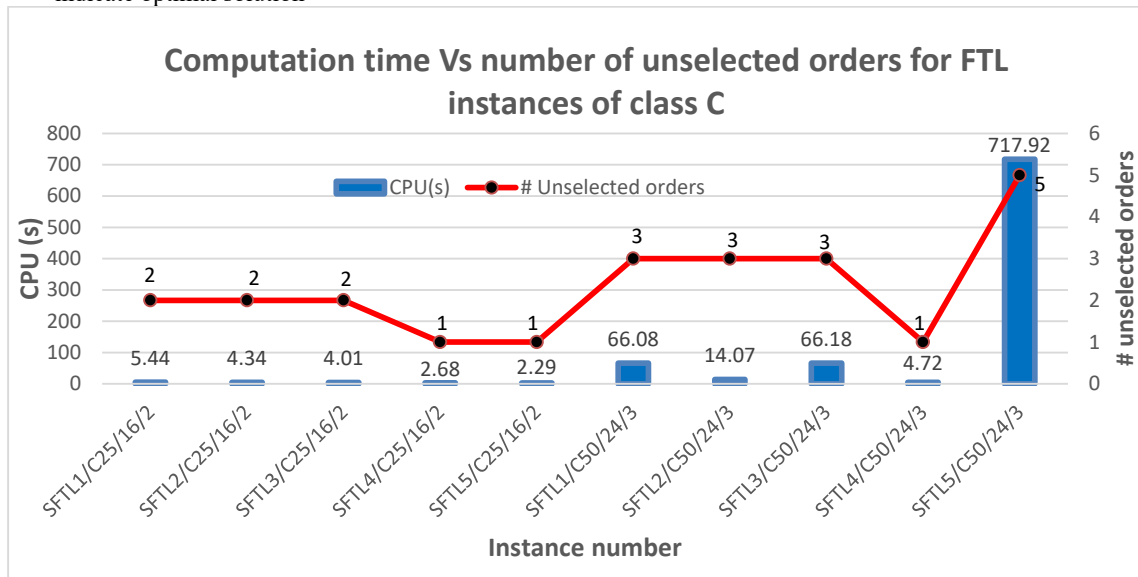
Fig. 3. Computation time Vs number of unselected orders for FTL instances of class C

Computation time Vs number of unselected orders for FTL instances of class R

Computation time Vs number of unselected orders for FTL instances of class RC

Fig. 3. Computation time vs. Number of unselected orders in the optimal solution from CPLEX for test instances $n = 24$ and $m = 3$ *(SFTL1 − 3/C50/24/3)* with a slight difference in CPU time that doesn't exceed 67 seconds, and the number of unselected orders is the same, that is 3. When $WTW = 180$ and $A_k^{max} = 600$, for $n = 16$ and $m = 2$, following the same pattern, the CPU times for the instances $SFTL4 − 5/C25/16/2$ are approximately the same and the number of unselected orders in the optimal solution is the same. For $n = 24$ and $m = 3$ *(SFTL4 − 5/C50/24/3)*, the CPU grows from 4.72 seconds, when the number of unselected orders is 1, to 717.92 seconds when the number of unselected orders is 5. This is explained by the fact that when the route duration of trucks is decreased, the number of unserved orders increases and subsequently, the CPU time increases.

For the **FTL** instances of class **R**, when $WTW = 120$ and $A_k^{max} = 480$, for $n = 20$ and $m = 2$; we have two different instances, the model solves the instance $SFTL2/R25/20/2$ in a fast CPU time that is 2.05 seconds, in which all orders are selected. As to the instance $SFTL1/R25/20/2$, there are two unselected orders and the CPU rises to 54 seconds. For $n = 30$ and $m = 3$, the $SFTL1/R50/30/3$ instance isn't solved optimally in which the number of unselected orders in the LB_{MILP} is 4 and the CPU is nearly 275 minutes. In contrast, the instance $SFTL2/R50/30/3$ is solved optimally by CPLEX in lot less CPU time and all orders are selected; it's a matter of bound. That is, in the first iterations, CPLEX finds the best bound. Maybe other parameters impact the quality of the solution and the CPU time. When $WTW = 180$ and $A_k^{max} = 720$, for $n = 20$ and $m = 2$ *(SFTL3 − 5/R25/20/2)*. The CPU time and the number of unselected orders are the same in all instances, where the first doesn't exceed 4 seconds and the latter is 1. For $n = 30$ and $m = 3$, even though the number of unselected orders is 1 for $SFTL3/R50/30/3$ and $SFTL4/R50/30/3$, CPU time explodes to some extent, perhaps the FTL instances of class R are the most difficult, especially that the CPU time of the instance $SFTL5/R50/30/3$ is nearly 162 minutes.

For the **FTL** instances of the class **RC**, when $WTW = 120$ and $A_k^{max} = 480$, for $n = 20$ and $m = 2$ *(SFTL1 −*

$2/RC25/20/2$), the CPU time in the instances is close and doesn't surpass 2 seconds, and the number of unselected orders is between 0 and 1. The CPU in this case is the same in the two classes C and R with the slightest difference depending on the number of unselected orders. For $n = 30$ and $m = 3$ ($SFTL1 - 2/RC50/30/3$), we have two instances, the CPU and the number of unselected orders are doubled from 66.97 seconds to 110.9 seconds and from 2 to 4, respectively. Class R is more challenging to solve than class RC; when the number of unselected orders is 4, the class R isn't solved optimally, and the class RC is solved optimally with a small CPU time (110.9 seconds). When $WTW = 180$ and $A_k^{max} = 720$, and when $n = 20$ and $m = 2$ ($SFTL3 - 5/RC25/20/2$), the CPU time of the instances ranges from 3 seconds to 12 seconds, depending on the number of unselected orders. The CPU time increases as the number of unselected orders increases. When $n = 30$ and $m = 3$ ($SFTL3 - 5/RC50/30/3$), it is noted that though the instances have the exact same numbers of unselected orders, there is a big difference between them concerning the CPU time, which is due to the complexity of RC. We observe that in case of 4 unselected orders, the CPU time in the class RC doesn't surpass 2289 seconds, whereas in the class R, the CPU is nearly 2h42 mins. Therefore, the class R is more difficult to solve than the class RC.

As expected, From Table 6 and Fig. 3, we notice that the CPU time goes up substantially with the increase of the number of orders, WTW, and especially the number of unselected orders in the optimal solution found by CPLEX. In all instances, when le the number of unselected orders increases, the trucks can't select some orders, so the CPU time certainly will increase. The selective aspect may be due to the following factors: the instance size, the geographic coordinates of the orders' pickup and delivery points, the geographical location of depots (see Fig. 4, Fig. 5 and Fig. 6), the loading and unloading TWs of demands and the time intervals $\left[D_k^{min}, A_k^{max}\right]$ during which the trucks are available. The results also confirm that $SFTL$ instances of the class R are the hardest to solve while class C instances are the easiest. In general, our proposed MILP model performs very well in terms of both solution quality and CPU time.
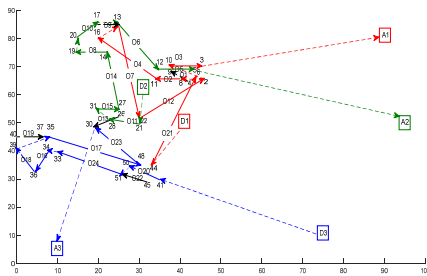

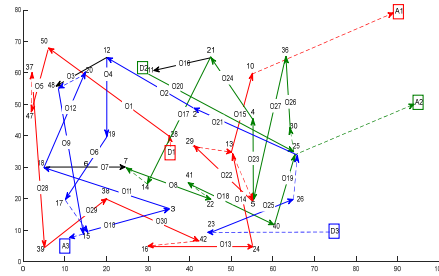**Fig. 4.** Result for instance $SFTL5/C50/24/3$

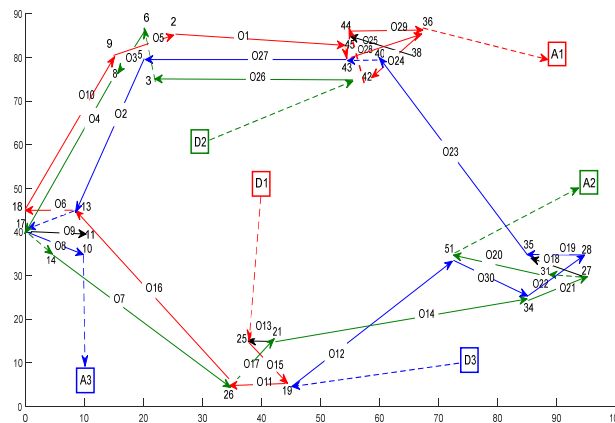**Fig. 5.** Result for instance $SFTL5/R50/30/3$

Fig. 6. Result for instance $SFTL5/RC50/30/3$

6. Conclusion

In this study, we have focused on an important vehicle routing planning, particularly an order selection and routing problem with full truckload, multiple depots and time windows (SFTMDVRPTW) in the context of an empty return scenario. We have proposed a mathematical formulation of the problem as a MILP model. The objective function is to maximize the total profit, which equals the total revenue from the selected orders minus the overall cost (cost of moving loaded trucks, cost of moving empty trucks, and cost of waiting). CPLEX 12.5 has been used to solve newly generated instances of up to 30 orders and 3 trucks. The results presented in this study are encouraging and demonstrate that the proposed MILP model provides plausible solutions; save for one instance, all other instances have been solved to optimality within an acceptable computing

time. We remark that the proposed model is strongly impacted by the selective aspect as well as the time window width of orders and depots. Avenues for future study are numerous. Firstly, it might be useful to study the sensitivity of the results when using a heterogeneous fleet of trucks or changing the model's input data to investigate the effect of these changes in the solution quality and CPU time. We also aim to improve this work by taking into account the provisions of the Labour Code, including driving times and rest periods. Our future research projects will study various facets of the SFTMDVRPTW, namely multi-objective SFTMDVRPTW. Secondly, since the problem is NP-hard, CPLEX is not able to solve relatively large instances. Therefore, we will develop an efficient meta-heuristic algorithm to solve the largest instances, focusing on the hybridization of genetic algorithm, ant colony optimization and tabu search.

References

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

Arunapuram, S., Mathur, K., & Solow, D. (2003). Vehicle Routing and Scheduling with Full Truckloads. *Transportation Science, 37*(2), 170-182.

Ávila, T., Corberán, Á., Plana, I., & Sanchis, J. M. (2017). Formulations and exact algorithms for the distance-constrained generalized directed rural postman problem. *EURO Journal on Computational Optimization, 5*(3), 339-365.

Ball, M. O., Golden, B. L., Assad, A. A., & Bodin, L. D. (1983). Planning for truck fleet size in the presence of a common-carrier option. *Decision Sciences, 14*(1), 103-120.

Braekers, K., Caris, A., & Janssens, G. K. (2013). Integrated planning of loaded and empty container movements. *OR Spectrum, 35*(2), 457-478.

Braekers, K., Caris, A., & Janssens, G. K. (2014). Bi-objective optimization of drayage operations in the service area of intermodal terminals. *Transportation Research Part E: Logistics and Transportation Review, 65*(1), 50-69.

Bettinelli, A., Ceselli, A., & Righini, G. (2011). A branch-and-cut-and-price algorithm for the multi-depot heterogeneous vehicle routing problem with time windows. *Transportation Research Part C: Emerging Technologies,19*(5), 723-740.

Bolaños, R. I., Escobar, J. W., & Echeverri, M. G. (2018). A metaheuristic algorithm for the multi-depot vehicle routing problem with heterogeneous fleet. *International Journal of Industrial Engineering Computations, 9*(4), 461–478.

Caris, A., & Janssens, G. K. (2009). A local search heuristic for the pre-and end-haulage of intermodal container terminals. *Computers and Operations Research, 36*(10), 2763-2772.

Currie, R. H., & Salhi, S. (2003). Exact and heuristic methods for a full-load, multi-terminal, vehicle scheduling problem with backhauling and time windows. *Journal of the Operational Research Society, 54*(4), 390-400.

Currie, R. H., & Salhi, S. (2004). A tabu search heuristic for a full-load, multi-terminal, vehicle scheduling problem with backhauling and time windows. *Journal of Mathematical Modelling and Algorithms, 3*(3), 225-243.

Desrochers, M., & Laporte, G. (1991). Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints. *Operations Research Letters, 10*(1), 27-36.

Desrosiers, J., Laporte, G., Sauve, M., Soumis, F., & Taillefer, S. (1988). Vehicle routing with full loads. *Computers and Operations Research, 15*(3), 219-226.

Dimitriou, L. (2021). Optimal competitive pricing in European port container terminals: A game-theoretical framework. *Transportation Research Interdisciplinary Perspectives, 9*, 1–12.

EL Bouyahyiouy, K., & Bellabdaoui, A. (2016). A new crossover to solve the full truckload vehicle routing problem using genetic algorithm. *2016 3rd International Conference on Logistics Operations Management (GOL)*, 1–6.

EL Bouyahyiouy, K., & Bellabdaoui, A. (2017). An ant colony optimization algorithm for solving the full truckload vehicle routing problem with profit. *2017 International Colloquium on Logistics and Supply Chain Management: Competitiveness and Innovation in Automobile and Aeronautics Industries, LOGISTIQUA 2017,7962888*, 142-147.

Golden, B. L., & Wong, R. T. (1981). Capacitated arc routing problems. *Networks, 11*(3), 305-315.

Grimault, A., Bostel, N., & Lehuédé, F. (2017). An adaptive large neighborhood search for the full truckload pickup and delivery problem with resource synchronization. *Computers and Operations Research, 88*, 1-14.

Gronalt, M., Hartl, R. F., & Reimann, M. (2003). New savings based algorithms for time constrained pickup and delivery of full truckloads. *European Journal of Operational Research, 151*(3), 520-535.

Gronalt, M., & Hirsch, P. (2007). Log-truck scheduling with a tabu search strategy.*Operations Research/ Computer Science Interfaces Series,39*, 65-88.

Imai, A., Nishimura, E., & Current, J. (2007). A Lagrangian relaxation-based heuristic for the vehicle routing with full container load. *European Journal of Operational Research, 176*(1), 87-105.

Jula, H., Dessouky, M., Ioannou, P., & Chassiakos, A. (2005). Container movement by trucks in metropolitan networks: Modeling and optimization. *Transportation Research Part E: Logistics and Transportation Review, 41*(3), 235-259.

Lahyani, R., Gouguenheim, A. -L., & Coelho, L. C. (2019). A hybrid adaptive large neighbourhood search for multi-depot open vehicle routing problems. *International Journal of Production Research, 57*(22), 6963-6976.

Lahyani, R., Khemakhem, M., & Semet, F. (2017). A unified matheuristic for solving multi-constrained traveling salesman problems with profits. *EURO Journal on Computational Optimization, 5*(3), 393-422.

Liu, R., Jiang, Z., Fung, R. Y. K., Chen, F., & Liu, X. (2010a). Two-phase heuristic algorithms for full truckloads multi-depot capacitated vehicle routing problem in carrier collaboration. *Computers and Operations Research, 37*(5), 950-

959.

Liu, R., Jiang, Z., Liu, X., & Chen, F. (2010b). Task selection and routing problems in collaborative truckload transportation. *Transportation Research Part E: Logistics and Transportation Review, 46*(6), 1071-1085.

Lysgaard, J., Letchford, A. N., & Eglese, R. W. (2004). A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming, 100*(2), 423-445.

Marín Moreno, C. A, Escobar Falcón, L. M, Bolaños, R. I, Subramanian, A., Escobar Zuluaga, A. H, & Echeverri, M. G. (2019). A hybrid algorithm for the multi-depot vehicle scheduling problem arising in public transportation. *International Journal of Industrial Engineering Computations, 10*(3), 361–374.

Nossack, J., & Pesch, E. (2013). A truck scheduling problem arising in intermodal container transportation. *European Journal of Operational Research, 230*(3), 666-680.

Rincon-Garcia, N., Waterson, B. J, & Cherrett, T. J. (2017). A hybrid metaheuristic for the time-dependent vehicle routing problem with hard time windows. *International Journal of Industrial Engineering Computations, 8*(1), 141–160.

Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research, 35*(2), 254-265.

Uchoa, E., Pecin, D., Pessoa, A., Poggi, M., Vidal, T., & Subramanian, A. (2017). New benchmark instances for the Capacitated Vehicle Routing Problem. *European Journal of Operational Research, 257*(3), 845-858.

Venkateshan, P., & Mathur, K. (2011). An efficient column-generation-based algorithm for solving a pickup-and-delivery problem. *Computers and Operations Research, 38(*12), 1647-1655.

Wang, X., & Regan, A. C. (2002). Local truckload pickup and delivery with hard time window constraints. *Transportation Research Part B: Methodological, 36*(2), 97-112.

Xue, N., Bai, R., Qu, R., & Aickelin, U. (2021). A hybrid pricing and cutting approach for the multi-shift full truckload vehicle routing problem. *European Journal of Operational Research, 292*(2), 500-514.

Yu, B., Yang, Z. Z., &Yao, B. Z. (2011). A hybrid algorithm for vehicle routing problem with time windows. *Expert Systems with Applications, 38*(1), 435-441.

Zhang, R., Yun, W. Y., & Moon, I. (2009). A reactive tabu search algorithm for the multi-depot container truck transportation problem. *Transportation Research Part E: Logistics and Transportation Review, 45*(6), 904-914.

Zhang, R., Yun, W. Y., & Kopfer, H. (2010). Heuristic-based truck scheduling for inland container transportation. *OR Spectrum, 32*(3), 787-808.

486