Computers and Chemical Engineering 63 (2014) 184–205

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

A monolithic approach to vehicle routing and operations scheduling of a cross-dock system with multiple dock doors

Rodolfo Dondo, Jaime Cerdá*

INTEC (UNL-CONICET), Güemes 3450, 3000 Santa Fe, Argentina

ARTICLE INFO

Article history:
Received 2 July 2013
Received in revised form
23 December 2013
Accepted 27 December 2013
Available online 19 January 2014

Keywords:
Cross-docking
Vehicle routing
Truck scheduling
MILP mathematical model
Sweeping-based approach

ABSTRACT

Cross-docking is a logistic strategy for moving goods from suppliers to customers via a cross-dock terminal with no permanent storage. The operational planning of a cross-dock facility involves different issues such as vehicle routing, dock door assignment and truck scheduling. The vehicle routing problem seeks the optimal routes for a homogeneous fleet of vehicles that sequentially collects goods at pickup points and delivers them to their destinations. The truck scheduling problem deals with the timing of unloading and reloading operations at the cross-dock. This work introduces a mixed-integer linear programming formulation for the scheduling of single cross-dock systems that, in addition to selecting the pickup/delivery routes, simultaneously decides on the dock door assignment and the truck scheduling at the cross-dock. The proposed monolithic formulation is able to provide near-optimal solutions to medium-size problems involving up to 70 transportation orders, 16 vehicles and 7 strip/stack dock doors at acceptable CPU times.

1. Introduction

Cross-docking is a logistic strategy used by many companies to decrease storage costs and improve customer satisfaction through a shorter delivery lead-time. The storage of goods is expensive because of space requirements, inventory holding costs and labor-intensive order picking tasks. Cross-docking seeks to eliminate a large portion of such warehousing costs. A cross-dock is usually an I-shaped facility with strip and stack dock doors located at opposite sides of the terminal and minimal storage space in between. Inbound shipments arriving at the cross-dock are allocated to strip docks on one side of the distribution terminal. Once the inbound trucks have been unloaded, the freights are screened and sorted by destination. After that, they are moved across the terminal via a forklift or a conveyor belt to their designated stack dock doors. There, the loads are charged into departing trucks carrying them to their destinations. Clearly, the handling of freight in a cross-dock terminal is a labor intensive and costly task because workers must unload, sort, and transfer a wide variety of loads from incoming trucks to outgoing trailers. Some products are better suited to cross-docking like (a) products having a stable demand; (b) perishable bulk materials, including some chemical and food compounds, requiring immediate shipment; (c) frozen foods and other refrigerated products like pharmaceuticals that should be directly moved from cooled inbound to cooled outbound trucks to keep the cooling chain unbroken; (d) high-quality items not needing quality inspection during the receiving process and (e) pre-tagged products that are ready for sale to the customers. Besides, drums of hazardous chemicals and containers of waste materials are usually aggregated at cross-dock facilities and immediately transferred to remedy sites for treatment and disposal. Pharmaceutical, food and chemical industries are increasingly using cross-docking to gain competitive advantages. Chemical and manufacturing companies like Eastman Kodak Co., Goodyear GB Ltd. and Toyota have reported the successful implementation of cross-docking strategies (Van Belle, Valckenaers, & Cattrysse, 2012).

Thorough reviews on cross-docking can be found in Boysen and Fliedner (2010) and Van Belle et al. (2012). Research was first focused on both the location and the physical layout of a cross-dock facility (e.g., the shape and the number of dock doors) and the related truck scheduling, but neglecting the routing aspects of the problem. The truck scheduling (TS) problem deals with operational issues at the cross-dock terminal that mainly include the assignment of vehicles to dock doors, the processing sequence of trucks at every strip and stack door and the transfer of goods from inbound to outbound vehicles. Although the idea of cross-docking is to unload inbound trucks and immediately reload the freights into delivery vehicles, a temporary storage is always necessary. Goods do not arrive at the cross-dock in the sequence they must be reloaded into the departing vehicles because a perfect synchronization of limited numbers of pickup and delivery trucks is impossible.

* Corresponding author. Tel.: +54 342 4559175; fax: +54 342 4550944.
E-mail address: jcerda@intec.unl.edu.ar (J. Cerdá).

0098-1354/$ – see front matter
http://dx.doi.org/10.1016/j.compchemeng.2013.12.012

An early work introducing a bilinear programming model to deal with the truck scheduling (TS) was due to Tsui and Chang (1992). The efficiency of workers depends in large part on the cross-dock layout and how trailers are assigned to dock doors. A good layout reduces transfer distances without creating congestion. Bartholdi and Gue (2000) introduced a layout design model that also considers the dock door assignment problem. The model is formulated to minimize transfer time, material handling and congestion through an efficient design of the cross-dock layout. Yu and Egbelu (2008) presented two approaches to schedule the trucks at the dock and find the better exchange of items between inbound and outbound trucks. The transfer of goods among trucks and the docking sequences of inbound and outbound trucks were simultaneously determined. Li, Low, Shakeri, and Lim (2009) also focused on truck scheduling and door assignment but considering a multi-door cross-dock and a number of trucks higher than the number of doors. Then, there will be lines of trucks waiting for an empty door to start onload/offload operations. Two approaches consisting of a mixed integer programming (MIP) model from a door scheduling viewpoint, and a dependency ranking search (DRS) heuristic algorithm were proposed. The problem goal was to minimize the total cross-dock operating time. The MILP model cannot be used for practical cases due to the high computational cost, while the DRS heuristic algorithm was able to find good solutions in much shorter solution times. Arabani, Ghomi, & Zandieh (2011) tested five different meta-heuristics such as the genetic algorithm (GA), the tabu search (TS), the particle swarm optimization (PSO), the ant-colony optimization (ACO) and the differential evolution (DE) algorithms, by applying them to solve a large number of cases. Chen et al. (2006) studied the truck scheduling problem for a network of cross-docks taking into consideration delivery and pickup time windows, warehouse capacities and inventory-handling costs. They solved the truck scheduling problem using local search techniques like simulated annealing and tabu-search and claimed that the heuristics outperform optimization models for providing good solutions in realistic time scales. Lee, Jung, and Lee (2006) developed an MILP formulation that considers both cross-docking operations and the vehicle routing problem, assuming that all vehicles coming from suppliers arrive at the cross-dock simultaneously in order to avoid vehicle waiting times at the cross-dock. Wen, Larsen, Clausen, Cordeau, and Laporte (2009) proposed a mixed integer programming formulation for the VRPCD problem involving pickup and delivery tasks to be started within specific time windows in order to minimize the traveled distance. The transportation requests are defined in terms of two locations: the pickup node where the freight is loaded and the delivery node to which is destined. Miao, Yang, Fu, and Xu (2012) studied a multi-crossdock transshipment problem with both soft and hard time windows. The flows from suppliers to customers via the cross-docks are constrained by fixed transportation schedules. Cargoes can be delayed and consolidated in cross-docks, and both suppliers and customers may alternatively have hard time windows or less-restrictive soft time windows. The formulation aims to minimize the total cost of multi-crossdock distribution networks, including transportation, inventory handling and penalty costs. As the problem is NP-hard, the authors proposed two solution methods based on meta-heuristics called Adaptive Tabu search and Adaptive Genetic algorithm, respectively. Dondo and Cerdá (2013) introduced a monolithic formulation for the VRPCD that determines the pickup and the delivery routes simultaneously with the truck scheduling at the cross-dock terminal, by assuming an unlimited number of dock doors. To get a computationally efficient approach, a set of constraints mimicking the widely known sweep heuristic algorithm (Gillet and Miller, 1974) to assign vehicles to pickup/delivery routes was incorporated into the MILP model. The sweep-heuristic based formulation can find near-optimal solutions to large problems at very acceptable CPU times. However, dock door assignments and queues of trucks in front of the dock doors were ignored.

If a limited number of dock doors is available, the assignment of them to incoming and outgoing trucks determines the efficiency of the cross-dock operations. In fact, a precise coordination among pickup vehicle routes, cross-dock activities and delivery vehicle routes is required to avoid long queues of trucks waiting for unloading/loading their cargoes. To this end, this work presents a new monolithic MILP formulation that integrates the pickup/delivery vehicle routing and scheduling with both the assignment of dock-doors to incoming and outgoing trucks and the managing of truck queues at strip/stack doors. Additional constraints mimicking the sweeping algorithm and avoiding symmetrical solutions are embedded into the mathematical formulation. In this manner, an efficient hybrid approach capable of solving medium-size problem instances at acceptable CPU times has been developed.

2. Problem definition

The combined vehicle routing and cross-dock truck scheduling problem (e.g., the VRPCD-TS problem) is defined as the problem of transporting a set of requests R from pickup to destination points passing through an intermediate cross-dock facility at minimum routing cost (see Fig. 1). The cross-dock is assumed to have a limited number of receiving (strip) doors RD and shipping (stack) doors SD. When an inbound (outbound) truck arrives (departs) at (from) the cross-dock, it must be decided to which dock door is assigned to increase the cross-dock productivity and reduce the handling cost. The truck scheduling (TS) problem seeks to find the optimal assignment of inbound/outbound trucks to dock doors. Most contributions on the VRPCD problem assumed that there are at least as much dock doors as trucks, so each truck will be assigned to a different door and truck scheduling aspects can be ignored. If this condition is not fulfilled, the dock doors can be seen as scarce resources that have to be scheduled over time. Lines of trucks waiting for service can arise at every dock door. This is the so-called truck scheduling problem. As the simultaneous treatment of both the VRPCD and the truck scheduling (TS) problems can be quite complex, they are usually solved in a sequential manner. In contrast to previous approaches, we will assume a limited number of dock doors and solve both the VRPCD and the TS problems at the same time.

The set of data to be considered in the formulation of the VRPCD-TS problem are next presented. Each transportation request $r \in R$ is described by specifying the shipment size q_r and the related pickup and destination locations. The Euclidean distance between pickup/delivery locations of requests $(r, r') \in R$, given by $d^P_{r,r'}/d^D_{r,r'}$ and the Polar coordinates ($r^P_{w,r}/r^D_{w,r}$ and θ^P_r/θ^D_r) of the pickup/delivery sites of request $r \in R$ (with the system origin at the cross-dock terminal) are also known data. The pickup and delivery tasks are fulfilled by the same set of homogeneous vehicles V each having a known capacity Q. Every vehicle departs from the cross-dock w, serves the assigned pickup locations and returns to the terminal for unloading the collected goods on the assigned receiving door. After completing offload operations, the vehicle moves to the shipping door of the terminal, reload orders and departs to their final destinations. The cross-dock terminal comprises given sets of receiving (RD) and shipping (SD) dock doors. The vehicle transfer-time between an inbound door $d \in RD$ and an outbound door $d' \in SD$ is given by the parameter $tt_{d,d'}$. The service time at each pickup/delivery location has two components: a fixed time for shipment-preparation (ft^P_r/ft^D_r) and a variable part that is proportional to the load size q_r. The loading/unloading rate at each pickup/delivery node is given by (lr_r/ur_r). Similar parameters for

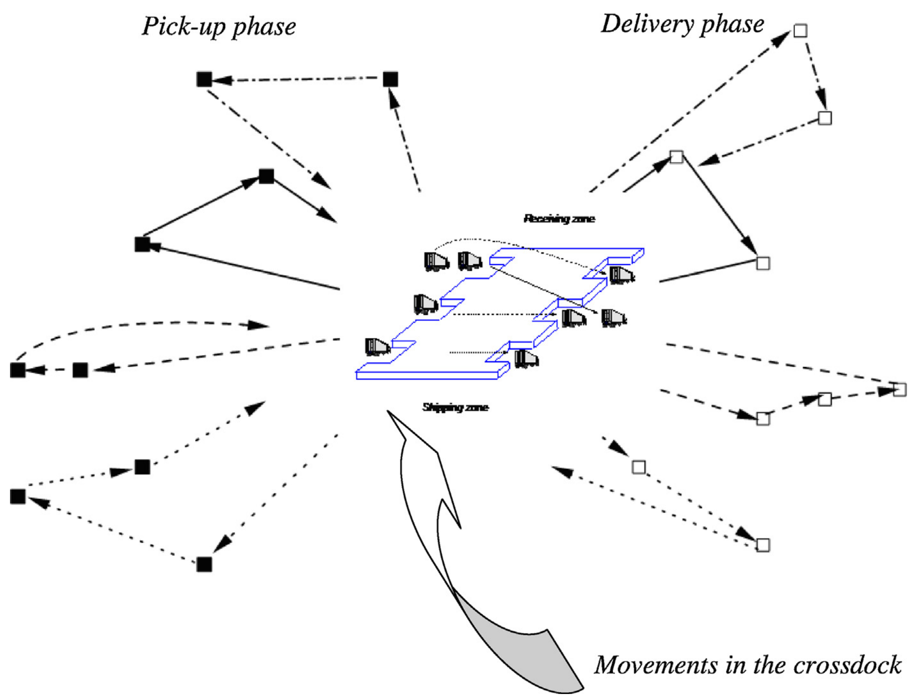

Fig. 1. Illustrating the VRPCD problem with a limited number of dock doors.

the cross-dock terminal are denoted by (ft_w^P/ft_w^D) and (lr_w/ur_w), respectively.

To generalize the mathematical formulation proposed by Dondo and Cerdá (2013) to simultaneously account for dock door assignment and the truck scheduling at the cross-dock, new binary variables are introduced to model the vehicle queues at the dock doors. On one hand, the binary variables $DP_{v,d}/DD_{v,d}$ have been defined to allocate vehicles to strip/stack dock doors. On the other hand, the relative ordering of trucks on the queues formed at the dock doors are controlled by the sequencing variables $ZP_{v,v'}/ZD_{v,v'}$. Moreover, the following new sets of continuous variables are to be defined: (a) the variable AT_v^P standing for the time at which the pickup vehicle v arrives at the cross-dock and waits for its turn in the truck queue of the assigned dock door to start unloading operations; (b) the variable RT_v^P denoting the time at which the pickup vehicle v is released from its pickup duties after completing the unloading tasks at the strip dock door; (c) the variables ST_v^P/ST_v^D representing the starting times of the pickup/delivery tours of vehicle v.

3. Model assumptions

The mathematical formulation has been developed based on the following assumptions:

(i) A homogeneous vehicle fleet transports goods from suppliers to destinations through a single cross-dock terminal.
(ii) The cross-dock has a known layout comprising a specific number of strip and stack dock doors.
(iii) All vehicles are available at the start of the planning horizon. They first accomplish the required pickup tasks and subsequently perform the delivery tasks.
(iv) Dock doors are exclusively dedicated to either unloading or loading operations, e.g. they are designated as either strip or stack dock doors.
(v) The number of strip/stack doors can be lower than the number of vehicles. Then, the dock doors can be regarded as scarce resources that should be scheduled over time.
(vi) Each P/D request must be serviced by a single vehicle, e.g. orders are not splittable.
(vii) The loading/unloading of a truck at the cross-dock cannot be interrupted, e.g. no pre-emption is allowed.
(viii) The freights unloaded at the cross-dock are not interchangeable, e.g. each one must be sent to a specific destination.
(ix) All activities must be completed within the given planning horizon t^{max}.
(x) The amounts of goods to be loaded or unloaded at supply/delivery locations are known data.
(xi) Each vehicle can service more than one pick-up/delivery location.
(xii) The pickup and delivery routes start and end at the cross-dock.
(xiii) The total quantity of goods carried by a vehicle must not exceed its capacity.
(xiv) The service time at supply/delivery locations is the sum of a fixed stop time (ft_r^P/ft_r^D) and a variable component directly increasing with the size of the cargo q_r to be picked-up/delivered at a rate lr_r/ur_r.
(xv) The goods picked up and delivered by the same truck are not unloaded at the cross-dock and remain inside the vehicle.
(xvi) The total amount of goods unloaded on the receiving docks and the total freight loaded on trucks at the shipping doors must be equal at the end of the planning horizon. So, there is no final inventory left at the cross-dock.

4. The milp mathematical model

4.1. Nomenclature

4.1.1. Sets

N	unload events
R	requests
RD	receiving (strip) dock doors
SD	shipping (stack) dock doors
V	vehicles

4.2. Binary variables

$DP_{v,d}/DD_{v,d}$ denotes that vehicle v has been allocated to the strip/stack dock door d
$WP_{n,v}/WD_{n,v}$ denotes that the unloading(U)/loading(L) activity of vehicle v is associated to the time event n
$XP_{r,r'}/XD_{r,r'}$ establishes the sequencing of pickup(P)/delivery(D) nodes (r, r') on the route of the assigned P/D vehicle
$YP_{r,v}/YD_{r,v}$ denotes that vehicle v visits the P/D location of request r
$ZP_{v,v'}/ZD_{v,v'}$ sequences vehicles (v, v') waiting for service at the same strip/stack door

4.3. Nonnegative continuous variables

$AT_v{}^P/AT_v{}^D$ P/D vehicle arrival times of vehicle v at the cross-dock facility
CP_r/CD_r Cumulative traveling cost from the cross-dock to the P/D site of request r
$DRS_{v,d,d'}$ denotes that the receiving door $d \in RD$ and the shipping door $d' \in SD$ have been assigned to vehicle v
$OC_v{}^P/OC_v{}^D$ overall traveling cost for the P/D tour of vehicle v
$RT_v{}^P$ time at which vehicle v is released from its pickup duties
$ST_v{}^P/ST_v{}^D$ starting time for the P/D tour of vehicle v
TP_r/TD_r vehicle arrival time at the P/D node of request r
TE_n unload time-event n
$UR_{r,n,v}$ denotes that request r was unloaded from vehicle v before or exactly at time TE_n
$UT_{r,n}$ denotes that the request r was unloaded on the cross-dock before or exactly at time event n
$YR_{r,v}$ states that the P/D locations of request r are both served by vehicle v

4.4. Parameters

$d^P{}_{r,r'}/d^D{}_{r,r'}$ distance between P/D locations of requests r and r'
$d^P{}_{r,w}/d^D{}_{r,w}$ distance between the P/D location r and the cross-dock w
$ft_r{}^P/ft_r{}^D$ fixed stop time at the P/D location of request r
$ft_w{}^P/ft_w{}^D$ fixed stop time for P/D activities at the cross-dock terminal w
lr_r/ur_r loading/unloading rate at P/D sites of request r
lr_w/ur_w loading/unloading rate at the cross-dock terminal w
q_r shipment size for request r
Q vehicle capacity
sp vehicle travel speed
t^{max} length of the planning horizon
$tt_{d,d'}$ time spent in moving a vehicle from the unloading door $d \in RD$ to the shipping door $d' \in SD$
uc_v unit distance cost for vehicle v

4.5. Decision variables for the sweeping-based constraints

$U^P{}_v/U^D{}_v$ denotes the existence of the pick-up/delivery sector for vehicle v

4.6. Positive variables for the sweeping-based constraints

ϕ_v^P lower angular limit of vth-pickup sector
$\Delta\phi_v^P$ angular width of the vth-pickup sector
ξ_r^P equals to one whenever the pickup location of request r satisfies the condition $\theta_r^P \in \left[0, \phi_1^P\right)$

4.7. Sweeping parameters

Δ maximum overlapping width between two adjacent sectors
$\Delta\theta$ angular distance used to pre-fix precedence relationships
Δd radial distance used to pre-fix precedence relationships

4.8. Model constraints

4.8.1. Route building constraints for the pick-up phase

Allocating requests to pickup vehicles. The pickup location of each request must be allocated to a single vehicle. If the assignment variable YP_{rv} is equal to 1, the pickup node of request r is served by the inbound vehicle v.

$$\sum_{v \in V} YP_{r,v} = 1 \quad \forall r \in R \tag{1}$$

Routing cost from the cross-dock up to the first visited node on a pickup route. Eq. (2) provides a lower bound on the routing cost from the cross-dock to any pickup node served by vehicle v (CP_r), including the first visited location. The parameter uc_v represents the routing cost per unit distance and $d_{w,r}^P$ denotes the distance between the cross-dock, identified by the subscript w, and the pickup site of request r.

$$CP_r \geq uc_v d_{w,r}^P YP_{r,v} \quad \forall r \in R \tag{2}$$

Cumulative routing cost from the cross-dock to a pickup node not visited on the first place. Sequencing constraints (3a) and (3b) relate the cumulative routing costs from the cross-dock to the pickup sites of a pair of requests $r,r' \in R$ served by the same vehicle v (i.e. $YP_{r,v} = YP_{r',v} = 1$). The formulation of such sequencing constraints uses a single binary variable $XP_{r,r'}$ (with $r < r'$) to select the relative order of any pair of pick-up nodes (r, r') located on the same inbound route. If $XP_{r,r'} = 1 (r < r')$, then the request r is served earlier than r'. By Eq. (3a), therefore, $CP_{r'}$ must be larger than CP_r by at least the routing cost along the path directly connecting both locations, i.e. the shortest route between the pickup sites of r and r'. Otherwise, $XP_{r,r'} = 0$ and node r' is visited before node r. Consequently, $CP_{r'}$ should be lower than CP_r by at least the cost term (uc_v $d_{w,r}^P$) by Eq. (3b). The parameter M_C^P is a relatively large number.

$$CP_{r'} \geq CP_r + uc_v d_{r,r'}^P - M_C^P (1 - XP_{r,r'}) - M_C^P (2 - YP_{r,v} - YP_{r',v}) \tag{3a}$$

$$CP_r \geq CP_{r'} + uc_v d_{r,r'}^P - M_C^P XP_{r,r'} - M_C^P (2 - YP_{r,v} - YP_{r',v}) \tag{3b}$$

$$\forall r, r' \in R \ (r < r'), \ v \in V$$

Overall routing cost for the tour assigned to pickup vehicle v. Every pickup route should end at the cross-dock facility. As the string of nodes on the route is unknown before solving the model, Eq. (4) provides a lower bound on the total routing cost for the vth-vehicle tour (OC_v^P) by assuming that any node on the route is the last visited. The largest bound determining the value of OC_v^P is set by the pickup location that is actually last visited by vehicle v.

$$OC_v^P \geq C_r^P + uc_v d_{r,w}^P - M_C^P (1 - YP_{r,v}) \quad \forall r \in R, \quad v \in V \tag{4}$$

Pickup node visiting times and vehicle arrival times at the cross-dock. Eqs. (5)–(7) allow determining both the visiting time for the pickup location r (TP_r) and the vth-vehicle arrival time (AT_v) at the cross-dock. Before performing the unloading operations, vehicle v should wait its turn on the queue of the assigned strip dock door. The timing constraints (6a)–(6b) present the same mathematical structures of Eqs. (3a) and (3b). They are indeed sequencing constraints involving routing time parameters instead of routing cost coefficients. The service time at any pickup node r is the sum of two terms: a fixed preparation time ft_r^P plus the variable loading time that directly increases with the load size q_r. The proportionality constant lr_r stands for the loading rate at the pickup node r. Moreover, the routing time along the path connecting the pickup nodes r and r' is given by the ratio between the distance $d_{r,r'}^P$ and the vehicle speed sp_v. The continuous variable ST_v^P stands for the starting time of the vth-pickup route. If all pickup routes are started at time $t = 0$, then $ST_v^P = 0$ for all $v \in V$.

$$TP_r \geq ST_v^P + \sum_{v \in V} \left(\frac{d_{wr}^P}{sp} \right) YP_{r,v} \quad \forall r \in R \tag{5}$$

$$TP_{r'} \geq TP_r + ft_r^P + lr_r\, q_r + \left(\frac{d_{r,r'}^P}{sp}\right) - M_T^P(1 - XP_{r,r'}) - M_T^P(2 - YP_{r,v} - YP_{r',v}) \tag{6a}$$

$$TP_{r'} \geq TP_r + ft_r^P + lr_r\, q_r + \left(\frac{d_{r,r'}^P}{sp}\right) - M_T^P\, XP_{r,r'} - M_T^P(2 - YP_{r,v} - YP_{r',v}) \tag{6b}$$

$$\forall r, r' \in R\ (r < r'),\ v \in V$$

$$AT_v^P \geq TP_r + ft_r^P + lr_r q_r + \left(\frac{d_{r,w}^P}{sp} \right) - M_T^P(1 - YP_{r,v}) \quad \forall r \in R, \quad v \in V \tag{7}$$

Vehicle capacity constraints. Eq. (8) states that the load transported by vehicle v cannot exceed its maximum capacity Q.

$$\sum_{r \in R} q_r YP_{r,v} \leq Q \quad \forall v \in V \tag{8}$$

4.8.2. Unloading operations at the receiving dock area

Identifying requests with pickup and delivery locations both served by the same vehicle. If the pickup and delivery sites of the request r are both served by the same vehicle, the related transshipment operations at the cross-dock are not required. In such a case, $YP_{rv} = YD_{rv} = 1$ for some vehicle v and the load of request r is not discharged on the receiving dock, i.e. it remains into the vehicle v. Let $YR_{r,v}$ be a non-negative continuous variable with a domain [0,1] that is defined to identify requests fully served by vehicle v. Eqs. (9)–(11) drives $YR_{r,v}$ to one whenever $YP_{r,v} = YD_{r,v} = 1$, and drops $YR_{r,v}$ to zero if either of such variables are null.

$$YR_{r,v} \leq YP_{r,v} \tag{9}$$

$$YR_{r,v} \leq YD_{r,v} \tag{10}$$

$$YR_{r,v} \geq YP_{r,v} + YD_{r,v} - 1 \tag{11}$$

$$\forall v \in V,\ r \in R$$

Allocating vehicles to receiving dock doors. By Eq. (12), a vehicle returning to the cross-dock from its pick-up trip must perform the unloading operations in just one receiving dock door $d \in RD$. Let us define the binary variable $DP_{v,d}$ to denote that the pickup vehicle v has been assigned to the strip dock door d whenever $DP_{v,d} = 1$. In Eq. (12), the set RD includes all the receiving doors available at the receiving dock.

$$\sum_{d \in RD} DP_{v,d} = 1 \quad \forall v \in V \tag{12}$$

Sequencing pickup vehicles assigned to the same strip dock door. The trucks leave the cross-dock after all freight has been unloaded. Eq. (13) defines a lower bound for the release time RT_v^P at which the pickup vehicle v completes the off-load operations at the cross-dock and is ready to perform delivery tasks. Such a bound is important for setting the value of RT_v^P for the vehicle first served at any receiving dock door. In turn, constraints (14a) and (14b) relate the times at which vehicles $(v, v') \in V$ $(v < v')$ end their unloading tasks just in case both vehicles have been assigned to the same strip door d $(DP_{r,v} = DP_{r',v} = 1)$. The relative order of a pair of vehicles v and v' on the queue of the common assigned door d is defined by a single variable $ZP_{v,v'}$ (with $v < v'$). If $ZP_{v,v'} = 1$, then vehicle v is served before. Otherwise, $ZP_{v,v'} = 0$ and truck v' is unloaded earlier. When the two vehicles are serviced at different strip dock doors, then the constraints (14a) and (14b) become redundant. The service time at every door is the sum of two components: a fixed preparation time (ft_W^P) and a variable service-time contribution that directly increases with the cargo to be unloaded given by $\sum_{r \in R} q_r\,(YP_{r,v} - YR_{r,v})$.

$$RT_v^P \geq AT_v^P + ft_w^P + ur_w \left[\sum_{r \in R} q_r\,(YP_{r,v} - YR_{r,v}) \right] \tag{13}$$

$$RT_{v'}^P \geq RT_v^P + ft_w^P + ur_w \left[\sum_{r \in R} q_r\,(YP_{r,v'} - YR_{r,v'}) \right] - M_T^P(1 - ZP_{v,v'}) - M_T^P(2 - DP_{v,d} - DP_{v',d}) \tag{14a}$$

$$RT_v^P \geq RT_{v'}^P + ft_w^P + ur_w \left[\sum_{r \in R} q_r\,(YP_{r,v} - YR_{r,v}) \right] - M_T^P\, ZP_{v,v'} - M_T^P(2 - DP_{v,d} - DP_{v',d}) \tag{14b}$$

$$\forall d \in RD, v, v' \in V\ (v < v')$$

Sequencing unloads events at the crossdock. An unload event n occurs at the cross-dock whenever a pickup vehicle v just completes the discharge of the cargoes to be delivered by other vehicles. Therefore, there will be as many unloads events in the set N as the number of pickup vehicles on duty. N is an ordered event set with the element n occurring before event n' $(n' > n)$. Let us define the binary variable $WP_{n,v}$ allocating pickup vehicles to unloads events, and the continuous variable TE_n representing the time at which the event n occurs. The event-time TE_n is set by the release time of vehicle v from its pickup assignments (RT_v^P) only if $WP_{n,v} = 1$. Eqs. (15a) and (15b) state that an inbound vehicle must exactly be assigned to a single time event and reciprocally an inbound vehicle can be allocated to only one event. Events assigned to unused vehicles will never occur, i.e. they are dummy events.

$$\sum_{n \in N} WP_{n,v} = 1 \quad \forall v \in V \tag{15a}$$

$$\sum_{v \in V} WP_{n,v} = 1 \quad \forall n \in N \tag{15b}$$

Moreover, Eq. (16a) indicates that event n takes place before event $n' > n$. Through Eq. (16a), the pickup vehicles should be assigned to unloads events in the same order that they complete their pickup duties. If the event n has been allocated to vehicle v $(WP_{n,v} = 1)$, then $TE_n = RT_v^P$. By Eq. (16b), the value of RT_v^P is imposed as a lower bound for TE_n whenever vehicle v has been assigned to either an earlier event $n' < n$ or to event n itself. The equality condition is forced by Eqs. (16c)–(16e).

$$TE_{n'} \geq TE_n \quad \forall n, n' \in N(n < n') \tag{16a}$$

$$TE_n \geq RT_v^P + M_T^P(WP_{n',v} - 1) \quad \forall n, n' \in N(n' < n), v \in V \tag{16b}$$

$$TE_n \leq RT_v^P \quad \forall n = first(N), v \in V \tag{16c}$$

$$RT_v^P \leq TE_{n'} + M_T^P(1 - WP_{n,v}) \quad \forall n, n' \in N(n < n') \tag{16d}$$

$$\sum_{n \in N} TE_n = \sum_{v \in V} RT_v^P \tag{16e}$$

Subset of requests already unloaded at the cross-dock at the event time TE_n. Let UR_{rnv} be a continuous variable with domain [0,1] denoting that request r collected by vehicle v is available for delivery on the cross-dock at the event time TE_n only if $UR_{rnv} = 1$. When the request r is not collected by vehicle v ($YP_{nv} = 0$) or is assigned to an event $n' \neq n$ ($WP_{nv} = 0$), Eqs. (17) and (18) drive UR_{rnv} to zero. If the reverse situation holds, UR_{rnv} is set equal to one by Eq. (19).

$$UR_{rnv} \leq WP_{n,v} \quad \forall n \in N, r \in R, v \in V \tag{17}$$

$$\sum_{n \in N} UR_{rnv} \leq YP_{r,v} \quad \forall r \in R, v \in V \tag{18}$$

$$UR_{rnv} \geq (WP_{n,v} + YP_{r,v} - 1) \quad \forall n \in N, r \in R, v \in V \tag{19}$$

The subset of requests already unloaded on the receiving dock at time TE_n is provided by the continuous variable $UT_{r,n}$ with domain [0,1]. If $UT_{r,n} = 1$, then the request r has been discharged from the pickup vehicle at a time earlier than or exactly at TE_n. In case the request r still remains on the cross dock at TE_n, it will be available for delivery at that time. The value of $UT_{r,n}$ is defined by Eq. (20).

$$UT_{rn} = \sum_{\substack{n' \in N \\ n' \leq n}} \sum_{v \in V} UR_{r,n',v} \quad \forall n \in N, r \in R \tag{20}$$

Usually, some loads are temporarily stored in front of the stack doors waiting for the arrival of the other goods to be also delivered by the assigned outbound truck.

Further queuing constraints for vehicles assigned to the same receiving door. When the inbound vehicles v and v' (with $v < v'$) have been allocated to the same receiving door $d \in RD$ and vehicle v features an earlier unload event ($WP_{nv} = 1$, $WP_{n'v'} = 1$ with $n < n'$), then by Eqs. (21a) and (21b) vehicle v must be served before v' and $ZP_{v,v'} = 1$. Otherwise, vehicle v' is unloaded before and $ZP_{v,v'} = 0$ by Eq. (21a). In case vehicles v and v' do not share the same strip dock door, Eqs. (21a) and (21b) become redundant.

$$ZP_{v,v'} \leq 2 - WP_{n,v} - \sum_{\substack{n \in N \\ n' < n}} WP_{n',v'} \tag{21a}$$

$$ZP_{v,v'} \geq WP_{n,v} + \sum_{\substack{n' \in N \\ n' > n}} WP_{n',v'} - 1 \quad \forall n \in N, v, v' \in V(v < v') \tag{21b}$$

4.8.3. Reloading operations at the shipping dock area

Allocating requests to outbound vehicles. As stated by Eq. (22), each transportation request must be allocated to a single outbound vehicle. Let us define the binary variable $YD_{r,v}$ to denote the allocation of request r to the outbound vehicle v only if $YD_{r,v} = 1$. Then,

$$\sum_{v \in V} YD_{r,v} = 1 \quad \forall r \in R \tag{22}$$

Allocating delivery vehicles to shipping dock doors. Let $DD_{v,d}$ be a binary variable allocating outbound vehicles to shipping doors. If $DD_{v,d} = 1$, then the loading operations for vehicle v will take place at the shipping door $d \in SD$. As stated by Eq. (23), an outbound vehicle on duty must be loaded at just one stack dock door. The set SD comprises the shipping doors available at the cross-dock.

$$\sum_{d \in SD} DD_{v,d} = 1 \quad \forall v \in V \tag{23}$$

Identifying the strip and stack dock doors assigned to each vehicle. The continuous variable $DRS_{v,d,d'}$ with domain [0, 1] has been introduced to indicate that vehicle v should move from the strip door $d \in RD$ to the stack door $d' \in SD$ before starting the loading operations. Eqs. (24a)–(24c) drive the variable $DRS_{v,d,d'}$ to one whenever $DP_{v,d} = DD_{v,d'} = 1$, and drops $DRS_{v,d,d'}$ to zero if either of such variables are null.

$$DRS_{v,d,d'} \leq DP_{v,d} \tag{24a}$$

$$DRS_{v,d,d'} \leq DD_{v,d'} \tag{24b}$$

$$DRS_{v,d,d'} \geq DP_{v,d} + DD_{v,d'} - 1 \tag{24c}$$

$$\forall v \in V, d \in RD, d' \in SD$$

Sequencing outbound vehicles assigned to the same shipping door. Let the continuous variable ST_v^D denote the time at which the delivery vehicle v starts the loading of the assigned requests at the cross-dock. Assuming that the same fleet of vehicles is used for pickup and delivery tasks, a pair of constraints should be considered on the value of ST_v^D: (a) the loading of a delivery vehicle v cannot start before completing its pickup assignments, i.e. not earlier than RT_v^P; and (b) the loading of vehicle v cannot begin until all the preceding trucks on the queue of the assigned stack dock door $d \in SD$ (i.e. $DD_{v,d} = 1$) have been served. Eq. (25) accounts for constraint (a) while Eqs. (26a) and (26b) mathematically describe the condition (b) by relating the times ST_v^D and $ST_{v'}^D$ at which the pair of vehicles $(v,v') \in V$ (with $v < v'$) assigned to the same shipping door d ($DD_{v,d} = DD_{v',d} = 1$) finish their loading activities at the cross-dock. If vehicle v precedes v' on the queue of door d, then the sequencing variable $ZD_{v,v'}$ is equal to one and Eq. (26a) applies. Otherwise, $ZD_{v,v'} = 0$ and Eq. (26b) becomes the relevant constraint. When the two vehicles are allocated to different stack dock doors, constraints (26a)–(26b) both become redundant. In Eqs. (25) and (26a)–(26b), the total loading time is equal to the sum of a fixed preparation time ft_w^D plus a variable time contribution that directly increases with the load size at a rate ur_w. Moreover, the time spent by a vehicle to move from the receiving door $d \in RD$ to the shipping door $d' \in SD$ is given by $tt_{d,d'}$. When the fleets of inbound and outbound vehicles are different, constraint (25) should be omitted. Then, the model can still be applied if the vehicles are either inbound or outbound trucks.

$$ST_v^D \geq RT_v^P + \sum_{d \in RD}\sum_{d' \in SD} tt_{d,d'}\, DRS_{v,d,d'} + ft_w^D + ur_w\left[\sum_{r \in R} q_r\,(YD_{r,v} - YR_{r,v})\right] \qquad v \in V \tag{25}$$

$$ST_{v'}^D \geq ST_v^D + ft_w^D + ur_w\left[\sum_{r \in R} q_r\,(YD_{r,v'} - YR_{r,v'})\right] - M_T^D\,(1 - ZD_{v,v'}) - M_T^D\,(2 - DD_{v,d} - DD_{v',d}) \tag{26a}$$

$$ST_v^D \geq ST_{v'}^D + ft_w^D + ur_w\left[\sum_{r \in R} q_r\,(YD_{r,v} - YR_{r,v})\right] - M_T^D\,ZD_{v,v'} - M_T^D\,(2 - DD_{v,d} - DD_{v',d})$$
$$\forall d \in SD, v, v' \in V\ (v < v') \tag{26b}$$

As the travel time between the docks is small compared with the time during which the freights should temporarily remain on the cross-dock, the constraint (25) is usually redundant.

Allocating delivery vehicles to unloads events. An outbound vehicle does not start loading operations until all the requests to be delivered by a truck are available at the cross-dock. This is so because the loading sequence is generally determined by: (a) the need of having the loads tightly packed into the truck and putting the fragile goods on the top, and (b) the ordering of the delivery nodes on the vehicle route (Van Belle et al., 2012). Let us define the binary variable $WD_{n,v}$ to denote that the outbound vehicle v has been assigned to the unload event $n \in ND$ only if $WD_{n,v} = 1$. The allocation of the outbound vehicle v to event n ($WD_{n,v} = 1$) means that the requests assigned to vehicle v ($YD_{r,v} = 1$) have already been unloaded on the cross-dock at a time earlier than or exactly TE_n. Such requests all feature $UT_{rn} = 1$ and, therefore, the condition $WD_{n,v} + YD_{r,v} = 2$ implies that $UT_{r,n} = 1$ and the loading of vehicle v cannot start before TE_n. Eq. (27) asserts that each outbound vehicle on duty must be assigned to a single unload event $n \in N$. However, several delivery vehicles can be allocated to the same unload event. Eq. (28) allows to make $WD_{n,v} = YD_{r,v} = 1$ only if the variable $UT_{r,n}$ is also equal to one. In this way, Eq. (28) prevents from allocating event n to an outbound vehicle v if $UT_{r,n} = 0$ for some request r with $YD_{r,v} = 1$. Moreover, by Eq. (29) an outbound vehicle v allocated to event n cannot start the loading operations before time TE_n. In addition, Eq. (30) drives the variable $WD_{n,v}$ to zero if the unload event for vehicle v occurs at some later event $n' > n$, i.e. $WP_{n',v} = 0$ for some $n' \leq n$. Eq. (30) should be omitted if every truck is either inbound or outbound.

$$\sum_{n \in N} WD_{n,v} = 1 \quad \forall v \in V \tag{27}$$

$$UT_{r,n} \geq (WD_{n,v} + YD_{r,v} - 1) \quad \forall n \in N, r \in R, v \in V \tag{28}$$

$$ST_v^D \geq TE_n + ft_w^D + ur_w\left[\sum_{r \in R} q_r(YD_{r,v} - YR_{r,v})\right] - M_T^D(1 - WD_{n,v}) \quad \forall n \in N, v \in V \tag{29}$$

$$WD_{n,v} \leq \sum_{\substack{n \in N \\ n' \leq n}} WP_{n',v} \quad \forall n \in N, v \in V \tag{30}$$

4.8.4. Route building constraints for the delivery phase

Constraint sets with mathematical structures similar to those proposed for the pickup phase can be written for delivery routes. Their formulations can be derived from Eqs. (2)–(8) by simply replacing the assignment variable $YP_{r,v}$ by $YD_{r,v}$, the routing cost CP_r by CD_r, the visiting time TP_r by TD_r, the sequencing variable $XP_{r,r'}$ by $XD_{r,r'}$ ($r < r'$), and the superscript P by D.

Outbound routing cost sequencing constraints. Sequencing constraints providing the outbound routing costs from the cross-dock up to the delivery site of request r are given by Eqs. (31)–(33). The parameter M_C^D is a relatively large number.

$$CD_r \geq \sum_{v \in V} uc_v d_{w,r}^D YD_{r,v} \quad \forall r \in R, v \in V \tag{31}$$

$$CD_{r'} \geq CD_r + uc_v\, d_{r,r'}^D - M_C^D\,(1 - XD_{r,r'}) - M_C^D\,(2 - YD_{r,v} - YD_{r',v}) \tag{32a}$$

$$CD_r \geq CD_{r'} + uc_v\, d_{r,r'}^D - M_C^D\, XD_{r,r'} - M_C^D\,(2 - YD_{r,v} - YD_{r',v}) \tag{32b}$$
$$\forall r, r' \in R\ (r < r'),\ v \in V$$

$$OC_v^D \geq CD_r + uc_v\, d_{r,w}^D - M_C^D\,(1 - YD_{r,v}) \quad \forall r \in R,\ v \in V \tag{33}$$

$$OC_v^D \geq CD_r + uc_v d_{r,w}^D - M_C^D(1 - YD_{r,v}) \quad \forall r \in R, v \in V \tag{34}$$

Vehicle stop times at delivery locations. The set of constraints providing lower bounds for the vehicle stop times at delivery locations are given by Eqs. (35)–(37).

$$TD_r \geq ST_v^D + \left(\frac{d_{w,r}^D}{sp_v}\right) - M_T^D(1 - YD_{r,v}) \quad \forall r \in R, v \in V \tag{35}$$

$$TD_{r'} \geq TD_r + ft_r^D + ur_r\, q_r + \left(\frac{d_{r,r'}^D}{sp_v}\right) - M_T^D\,(1 - XD_{r,r'}) - M_T^D\,(2 - YD_{r,v} - YD_{r',v}) \tag{36a}$$

$$TD_r \geq TD_{r'} + ft_{r'}^D + ur_{r'}\, q_{r'} + \left(\frac{d_{r',r}^D}{sp_v}\right) - M_T^D\, XD_{r,r'} - M_T^D\,(2 - YD_{r,v} - YD_{r',v}) \tag{36b}$$
$$\forall r, r' \in R\ (r < r'),\ v \in V$$

$$AT_v^D \geq TD_r + ft_r^D + ur_r q_r + \left(\frac{d_{r,w}^D}{sp_v}\right) - M_T^D(1 - YD_{r,v}) \quad \forall r \in R, v \in V \tag{37}$$

Vehicle capacity constraints. The load to be transported by a delivery vehicle cannot exceed its maximum capacity Q.

$$\sum_{r \in R} q_r YD_{r,v} \leq Q \quad \forall v \in V \tag{38}$$

Further queuing constraints for vehicles sharing the same shipping door. If delivery vehicles v and v' are loaded at the same stack dock door and vehicle v has been assigned to an earlier event, then vehicle v is served before and $ZD_{v,v'} = 1$ by Eq. (39). In the reverse case, vehicle v' is loaded earlier and $ZD_{v,v'} = 0$. When the vehicles have been allocated to different shipping doors, the value of $ZD_{v,v'}$ is meaningless.

$$\left\{ \begin{array}{l} ZD_{v,v'} \leq 2 - WD_{n,v} - \sum\limits_{\substack{n' \in N \\ n' < n}} WD_{n',v'} \\ ZD_{v,v'} \geq WD_{n,v} + \sum\limits_{\substack{n' \in N \\ n' > n}} WD_{n',v'} - 1 \end{array} \right\} \quad \forall n \in N, v, v' \in V (v < v') \tag{39}$$

4.8.5. Valid inequality constraints

Additional constraints relating the total routing cost and the vehicle arrival times usually help to speed up the solution process.

Valid inequalities for the pickup phase. By relating the arrival time AT_v^P and the total routing cost for the pickup tour of vehicle v, lower and upper bounds on the value of AT_v^P can be obtained through Eqs. (40a) and (40b), respectively. Such bounds are obtained by estimating AT_v^P as the sum of the starting time ST_v^P plus the total service time at the visited locations and the total traveling time. If time windows for the service start at the P/D locations are not considered, the parameter η is equal to zero. Nonetheless, it has been chosen $\eta_P = 0.001$ to account for round off errors. For problems with narrow time windows, the value of η_P should be increased to 0.1–0.3 because sometimes the pickup vehicles should wait for the opening of the time window at some visiting sites.

$$AT_v^P \geq (1-\eta_P) \quad ST_v^P + \left(\frac{OC_v^P}{uc_v sp_v} \right) + \sum_{r \in R} \left(ft_p + lr_r\, q_r \right) YP_{r,v} \qquad v \in V \tag{40a}$$

$$AT_v^P \leq (1+\eta_P) \quad ST_v^P + \left(\frac{OC_v^P}{uc_v sp_v} \right) + \sum_{r \in R} \left(ft_p + lr_r\, q_r \right) YP_{r,v} \qquad v \in V \tag{40b}$$

Valid inequality constraints for the delivery phase. Constraints (41a)–(41b) that are similar to Eqs. (40a) and (40b) are proposed for the delivery phase.

$$AT_v^D \geq (1-\eta_D) \quad ST_v^D + \left(\frac{OC_v^D}{uc_v sp_v} \right) + \sum_{r \in R} (ft_r^D + ur_r\, q_r)\, YD_{r,v} \qquad v \in V \tag{41a}$$

$$AT_v^D \leq (1+\eta_D) \quad ST_v^D + \left(\frac{OC_v^D}{uc_v sp_v} \right) + \sum_{r \in R} (ft_r^D + ur_r\, q_r)\, YD_{r,v} \qquad v \in V \tag{41b}$$

Valid inequality constraints for the allocation of receiving dock doors to vehicles. To partially eliminate symmetric solutions, constraints (42) are incorporated into the mathematical model just to solve large problems. If the set RD comprises three elements $\{rd_1, rd_2, rd_3\}$, then constraints (42) allocates the dock door rd_1 to the vehicle v^* that first unloads the cargo on the cross-dock terminal (e.g., $WP_{n1,v^*} = 1$), the dock door rd_2 to the vehicle $v^\#$ completing the unloading operations in the second place (e.g., $WP_{n2,v\#} = 1$) and rd_3 to the truck finishing the pickup duties on third place. Constraints (42) do not exclude the optimal solution from the feasible region but just avoid symmetrical assignments.

$$\sum_{\substack{rd \in RD \\ rd < n}} DP_{v,rd} \geq \sum_{\substack{n' \in N \\ n' < n}} WP_{n',v} \quad \forall v \in V, n \in N (n \leq |RD|) \tag{42}$$

4.9. The objective function

Depending on the relative sizes of the major costs involved in the problem, alternative objective functions can be used.

(a) Minimizing the cumulative vehicle routing cost

$$\text{Min} \quad z = \sum_{v \in V} \left[(OC_v^P + OC_v^D) \right] \tag{43a}$$

(b) Minimizing the cumulative distribution time

$$\text{Min} \quad z = \sum_{v \in V} AT_v^D \tag{43b}$$

(c) Minimizing the total makespan

$$\text{Min} \quad z = MK \text{ with } MK \geq AT_v^D \quad \forall v \in V \tag{43c}$$

The objective functions (43b) and (43c) both seek to reduce the total distribution time. However, the MILP solution algorithm shows a better computational performance with the objective function (43b). This is why it was selected to solve the examples in Section 5 when the problem goal is to minimize the total distribution time.

(d) Minimizing a weighted combination of objectives (a) and (b)

$$\text{Min} \quad z = \mu \sum_{v \in V} AT_v^D + \sum_{v \in V} \left[(OC_v^P + OC_v^D) \right] \tag{43d}$$

In Eq. (43d) the coefficient μ represents the cost per unit time spent in fulfilling the pickup and delivery tasks.

5. Handling the computational np-hardness

The proposed mathematical formulation shows an exponential increase of the solution time with the number of transportation requests. An important fraction of the computational burden is associated to two major tasks: (1) assigning vehicles to pickup/delivery nodes, and (2) sequencing P/D locations on a vehicle tour. Both problems will be separately attacked to alleviate the intrinsic NP-hardness of the problem.

5.1. Allocating vehicles-to-nodes using the sweeping algorithm

The sweeping algorithm (Gillet and Miller, 1974) is a heuristic technique that efficiently solves the VRP problem with a single depot and a homogeneous fleet of vehicles. The depot is at the origin of a polar coordinate system through which each location is described in terms of the radial ($d_{w,r}$) and the angular (θ_r) coordinates. For the vehicle assignment process, the customer nodes are arranged by increasing θ_r and the nodes are assigned to the current vehicle as the angular coordinate continually rises while it is not overloaded. Otherwise, a new vehicle is chosen and the procedure is continued until every site has been assigned to exactly one vehicle. Dondo and Cerdá (2013) introduced a set of equations shown in Appendix A to mimic the sweeping algorithm. By including them into the mathematical formulation, they can solve the VRPCD problem with an unlimited number of receiving/shipping dock doors. In other words, it was assumed that every pickup/delivery vehicle is assigned to a different dock door and serviced without delay. By applying the sweeping-based constraints, P/D sites of the requests are grouped into a number of angular sectors each one assigned to a different vehicle. The width of the angular areas is adjusted in order to minimize the value of the objective function. Moreover, the best polar angle for starting the procedure is also optimized (see Appendix A).

5.2. Breaking the driving direction symmetry

The travel direction of a tour can be predefined by specifying in which order the pickup locations of two requests (r_1, r_2) along the route will be visited. This can be done by setting the value of the related sequencing variable $XP_{r1,r2}$ to either one or zero, assuming that $r_1 < r_2$. Though the procedure is explained for pickup routes, it can also be applied to fix the driving direction for delivery tours. In general, the angular coordinates of the pickup locations of two requests belonging to the same tour are rather close, especially if the number of pickup tours increases. So, we should focus on fixing the value of $XP_{r1,r2}$ for pairs of nodes with angular coordinates not too dissimilar. However, it is intuitively desirable to fix as few precedence relationships as possible. If the values of $(\theta^P_{r1}, \theta^P_{r2})$ stand for the angular coordinates of the pickup sites of requests r_1 and r_2, then the difference $\left|\theta^P_{r1} - \theta^P_{r2}\right|$ must be greater than a certain angular limit $\Delta\theta$ lowering with the number of tours (condition A). In this way, the visiting order of few pairs of requests (r_1, r_2) served by the same vehicle will be predefined by setting the value of $XP_{r1,r2}$ to either zero or one. At the same time, a significant model size reduction is obtained because many irrelevant sequencing variables $XP_{r1,r2}$ associated to requests allocated to different tours are also fixed. The angular limit $\Delta\theta$ has been adopted using the following expression: $\Delta\theta = 2\pi/\gamma_\theta \left|V\right|$, where $|V|$ stands for the number of pickup vehicles and γ_θ is a parameter whose value must be properly tuned in order to control the number of frozen precedence relationships. In this work, it was chosen $\gamma_\theta = 3$–4. Moreover, the driving direction constraints should be rather lax to avoid cutting portions of the feasible solution space. Then, the pair of nodes with prefixed values of $XP_{r1,r2}$ must not be too close to still allow the tour to take alternative configurations. In other words, we are interested in pairs of nodes not only satisfying condition (A) but also having dissimilar radial coordinates. To prefix the ordering of requests (r_1, r_2) on a pickup vehicle route, $\left|d^P_{w,r1} - d^P_{w,r2}\right|$ must be greater than certain limit Δd (condition B). Parameters $(d^P_{w,r1}, d^P_{w,r2})$ denote the radial coordinates of the pickup sites of requests r_1 and r_2. The value of Δd must be carefully chosen so that the relative ordering of few pairs of nodes are prefixed. In this work, Δd has been determined using the following criterion: $\Delta d = \Delta d_{\max}/\gamma_d$, with $\Delta d_{\max} = \max\limits_{\substack{r1, r2 \in R \\ r1 < r2}} \left|d^P_{w,r2} - d^P_{w,r1}\right|$ and γ_d is a parameter whose value should be properly tuned. In this work, it has been adopted $\gamma_d = 4$–5. Given the pickup location of a request r_1, a procedure should be developed to identify the candidate pairs of requests (r_1, r_2), with $r_1 < r_2$ satisfying simultaneously both conditions (A) and (B): $\left|\theta^P_{r1} - \theta^P_{r2}\right| > \Delta\theta$ and $\left|d^P_{w,r1} - d^P_{w,r2}\right| > \Delta d$. Let us assume that all the vehicles must travel counter clockwise. Given r_1, the candidate pairs (r_1, $r_2 < r_1$) for pre-fixing the values of their related variables $XP_{r1,r2}$ are those fulfilling the following two conditions: $\theta^P_{r2} > (\theta^P_{r1} + \Delta\theta)$ and $\left|d^P_{w,r1} - d^P_{w,r2}\right| > \Delta d$. More precisely, a counter clockwise driving direction can be fixed by setting the variables $XP_{r1,r2}$ to one if $\theta^P_{r2} > (\theta^P_{r1} + \Delta\theta)$ and $d^P_{w,r2} > (d^P_{w,r1} + \Delta d)$, or

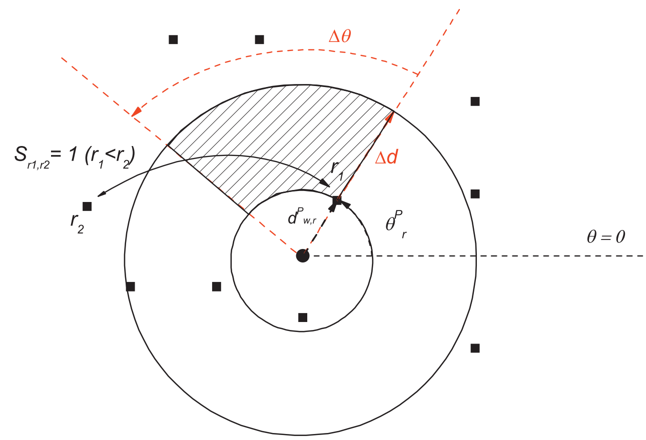

Fig. 2. Parameters Δd and $\Delta\theta$ defining the nodes to preorder with regards to *r1*.

to zero if $\theta^P_{r2} > (\theta^P_{r1} + \Delta\theta)$ and $d^P_{w,r1} > (d^P_{w,r2} + \Delta d)$. Such conditions are given by the following constraints:

$$
\begin{aligned}
&(d^P_{w,r2} > d^P_{w,r1} + \Delta d) \wedge (\theta^P_{r2} > \theta^P_{r1} + \Delta\theta) \Rightarrow X^P_{r1,r2} = 1 \\
&(d^P_{w,r1} > d^P_{w,r2} + \Delta d) \wedge (\theta^P_{r2} > \theta^P_{r1} + \Delta\theta) \Rightarrow X^P_{r1,r2} = 0 \\
&\qquad \forall r1, r2 \in R(r1 < r2)
\end{aligned}
$$

In Fig. 2, it is shown an angular sector with a width $\Delta\theta$ generated by a ray connecting the cross-dock with the location of request r_1 and moving counter clockwise. Candidate pairs of nodes (r_1,r_2) for fixing the values of the related variables $XP_{r1,r2}$ to one include just those nodes r_2 which are outside of the shaded angular sector and feature a radial coordinate $d^P_{w,r2} > (d^P_{w,r1} + \Delta d)$. When $d^P_{w,r1} > (d^P_{w,r2} + \Delta d)$, a counter clockwise direction is predefined by setting $XP_{r1,r2} = 0$.

6. Results and discussion

To illustrate the effectiveness of the proposed formulation on providing high-quality solutions to medium-size VRPCD-TS problems within bounded CPU times, a sizable number of new case studies introduced in this work has been solved. The total vehicle routing cost and the total distribution time ($\sum_{v \in V} ATD_v$) were selected as the objective functions to be minimized. The same vehicle fleet is used to visit pickup and delivery request locations. Several instances of two medium-size examples have been solved. Example 1 involves up to 46 transportation orders while Example 2 accounts for at most 70 requests. The data for the transportation requests of both examples, including the shipment size, the Cartesian coordinates of the related P/D locations and the time windows for starting operations at P/D nodes are reported in Table B.1 of Appendix B and Table C.1 of Appendix C, respectively. Several problem instances involving different cross-dock layouts were generated by considering the first *N* requests listed in Tables B.1 and C.1 with N varying from 8 to 46 for Example 1, and from 50 to 70 for Example 2. Each problem instance is labeled by the number of requests $|N|$, the number of available vehicles $|V|$, and the number of strip doors $|RD|$ and stack dock doors $|SD|$. The vehicle capacity is equal to 75 units for most instances of Example 1 and 90 units for all instances of Example 2. The vehicle capacity is reduced from 75 to 50 units for the three largest instances of Example 1. The vehicle transfer times between strip and stack dock doors for Examples 1 and 2 are reported in Table B.2 of Appendix B and Table C.2 of Appendix C, respectively. The selected values for the model parameters in all problem instances of Examples 1 and 2 are given by: $lr_r = ur_r = 0.2$; $ft^P_r = ft^D_r = 0.5$; $lr_w = ur_w = 0.5$ and $ft^P_w = ft^D_w = 0.5$. The customer requests must be fulfilled within the planning horizon

Table 1
Best solutions for small instances of Example 1 using the exact formulation.

Example	|N|	|V|	|RD|	|SD|	Best solution	Gap (%)	CPU (s)
1	8	2	2	2	295.1	–	90.6
2	9	2	2	2	329.0	–	101.4
3	10	2	2	2	398.6	–	392.1
4	11	3	2	2	414.3	17.6	3600[a]
5	12	3	2	2	473.2	26.6	3600[a]

[a] CPU time limit.

Table 2
Best solutions for small instances of Example 1 using the hybrid formulation.

Example	|N|	|V|	|RD|	|SD|	Δ	Best solution	Gap (%)	CPU (s)
1	8	2	2	2	0.	295.1	–	10.2
2	9	2	2	2	0.	329.0	–	5.2
3	10	2	2	2	0.	398.6	–	70.5
4	11	3	2	2	0.	422.8	–	69.9
					0.5	414.3	–	3553.8
5	12	3	2	2	0.	477.9	–	257.1
					0.5	473.2	–	942.1

going from $t = 0$ to $t^{max} = 400$ time units. When time windows are imposed on the service start time, the value of t^{max} is increased to 450. All problem instances were run on GAMS 23.7 using a 2.66 MHz two-processor PC with 24 MB RAM and 6 cores-per-processor. The relative gap tolerance was set at 10^{-2} and a maximum CPU time of 3600 s is allowed for all instances of Example 1. The CPU time limit was increased to 5000 s for all instances of Example 2. If the problem cannot be solved to optimality within the time limit, the best integer solution found and the related integrality gap are both reported.

6.1. Validating the proposed hybrid MILP formulation

The proposed exact formulation for the VRP-TS problem (obtained from ignoring the sweeping-based constraints and the symmetry breaking rule) is able to just solve small examples due to the high complexity and the inherent NP-hardness of the problem. Nevertheless, five small problem instances of Example 1 were tackled with both the exact and the hybrid VRP-TS formulations to compare their best solutions. In this way, we can verify the quality of the solutions provided by the hybrid representation and the CPU time saving achieved using the inexact approach. The minimum total pickup/delivery (P/D) routing cost has been selected as the primary problem goal and the total distribution time ($\sum_{v \in V} ATD_v$) was chosen as the secondary goal. After finding the minimum routing cost target, the P/D routes are fixed and the model is solved again to minimize the total P/D time. Table 1 reports the minimum routing-cost solutions to five small problem instances of Example 1 using the exact formulation, while Table 2 shows the best solutions discovered by the hybrid model. The selected values for the tuning parameters of the hybrid model were $\Delta = 0$ (no overlapping is allowed), $\Delta d = max_r\ d_{w,r}/3$, and $\Delta\theta = 2\pi/(2|V|)$. In all instances, the cross-dock layout includes two strip and two stack dock doors.

Table 3
Minimum cost solution for the instance 12R-3V-2RD-2SD of Example 1.

Pick-up routes

Vehicle	Tour	Load Collected	Arrival time	Tour cost
V1	r1-r9-r3-r12	58	87.8	74.2
V2	r6-r5-r8-r11-r7-r10	71	110.0	92.8
V3	r2-r4	26	69.4	63.2

Unloading operations

Receiving dock door	Vehicle	Service start time	Drop-off requests	Vehicle leaving time
RD1	V1	87.8	r1-r9-r3-r12	117.3
RD2	V3	69.4	r4	79.4
	V2	110.0	r6-r5-r8-r11-r7-r10	146.0

Shipping operations

Shipping dock door	Vehicle	Vehicle arrival time	Ship-on requests	Service start-time	Service completion time
SD1	V3	83.4	r1-r3-r9-r12-r6	146.0	177.0
SD2	V2	148.0	r4	148.0	158.0
	V1	121.3	r5-r7-r11-r10-r8	158.0	192.5

Delivery routes

Vehicle	Load to deliver	Vehicle departure time	Tour	Vehicle arrival time	Tour cost
V1	68	192.5	r5-r7-r11-r10-r8	295.4	86.9
V2	19	158.0	r4	172.3	10.0
V3	68	177.0	r1-r3-r9-r12-r2-r6	339.7	146.1
Total P/D vehicle routing cost					473.2

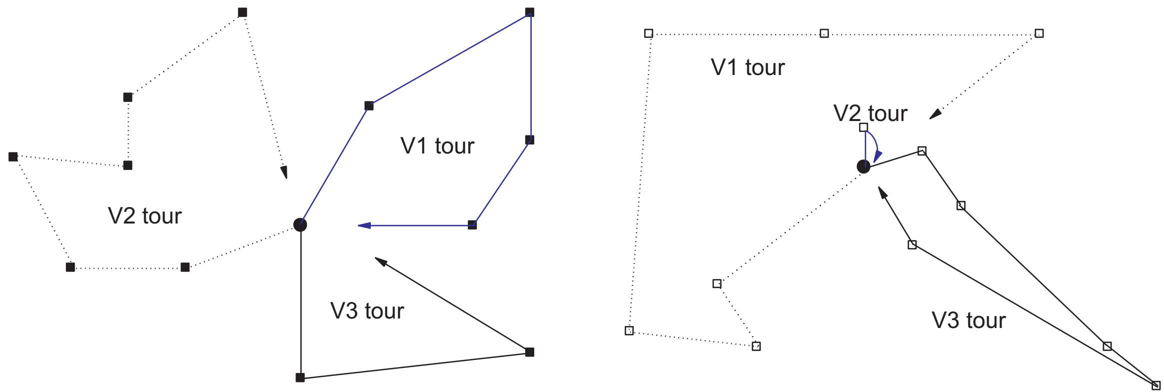

Fig. 3. Minimum cost solution for the instance 12R-3V-2RD-2SD of Example 1.

The hybrid formulation was able to find the best solution provided by the exact model for the first three problem instances: [8R-2V-2RD-2SD], [9R-2V-2RD-2SD] and [10R-2V-2RD-2SD]. In the other two instances of Example 1, e.g. [11R-3V-2RD-2SD] and [12R-3V-2RD-2SD], the best solutions present a sub-optimality level of 2.0% and 1.05% with regards to the ones provided by the exact formulation, respectively. Nevertheless, the truly optimal solutions for these two examples were found in 3553 s and 942 s, respectively, by adopting $\Delta = \pi/6$ to allow route overlapping. In both examples, the exact formulation was unable to prove the solution optimality within the CPU time limit.

Sketches of the pickup and delivery tours for the instance 12R-3V-2RD-2SD of Example 1 are shown in Fig. 3. It can be observed the presence of a complete route overlapping between the delivery tours of vehicles V1 and V2. Besides, Table 3 gives complete information on the best P/D vehicle tours for the instance 12R-3V-2RD-2SD of Example 1 including the sequence of vehicle stops, the collected load, the P/D vehicle arrival times at the cross-dock and the tour routing costs. Besides, it reports the assignment of dock doors to P/D vehicles and a detailed schedule of unloading and reloading operations at the cross-dock facility. The limited number of stack dock doors produces a delay in the loading of vehicle V1 at SD2.

6.2. *Minimizing the total P/D vehicle routing cost for larger instances of Example 1*

The VRPCD-TS hybrid formulation is now applied to a series of 20 problem instances of Example 1 involving from 14 to 46 requests and different cross-dock layouts. The problem goal is to minimize the total P/D vehicle routing cost. Moreover, the least total distribution time ($\sum_{v \in V} ATD_v$) was chosen as the secondary target to be achieved by solving again the VRPCD-TS model after freezing the optimal P/D vehicle routes. To do that, the binary variables YP, YD, XP and XD defining the P/D routes are fixed at their optimal values before resolving the problem again. In other words, the cost-based and the time-based targets are sought in a sequential manner. Table 4 reports the best solutions found using the hybrid formulation. The selected values for the model parameters were $\Delta = 0$, $\Delta d = (max_r\ d_{w,r})/3$, and $\Delta\theta = 2\pi/(2|V|)$ for all instances of Example 1.

As shown in Table 4, the sweeping-based hybrid formulation is able to solve almost all problem instances of Example 1 with up to 46 requests and 5 inbound/outbound dock doors to optimality in reasonable CPU times. The optimality gap always remains below 1.6% within the time limit of 3600 s. Moreover, the best solution is often discovered in a reasonable CPU time. As expected, changes in

Table 4
Minimum cost solutions for instances of Example 1 involving 10–46 requests.

|N|	|V|	|RD|	|SD|	Best solution	Gap (%)	CPU time (s)	
						To find the best solution	To prove optimality
14	3	1	1	553.2	–		107.9
16	4	2	2	628.1	–		145.5
18	4	1	1	655.9	–		348.4
		2	2	655.9	–		307.5
20	4	2	2	774.4	–		60.4
		3	3	774.4	–		67.9
22	5	3	3	805.5	–		1492.8
24	5	2	2	909.8	–	124.0	2445.0
		3	3	909.8	–		1908.6
26	6	3	3	928.9	–	317.0	1397.0
28	6	3	3	952.0	–	399.0	1502.9
30	6	3	3	987.2	–	176.0	356.0
32	7	3	3	1003.3	1.52	3552.0	3600*
34	7	3	3	1052.3	1.08	3231.0	3600*
36	7	3	3	1121.9	–	1162.0	1365.0
38	8	3	3	1231.6	–	2934.0	3275.7
40	8	3	3	1329.1	–	614.0	1537.9
42	11	5	5	1693.9	–	1761.1	3017.8
44	12	5	5	1739.8	–	2681.5	3441.0
46	12	4	4	1863.1	–	3082.2	3532.5

* CPU time limit.

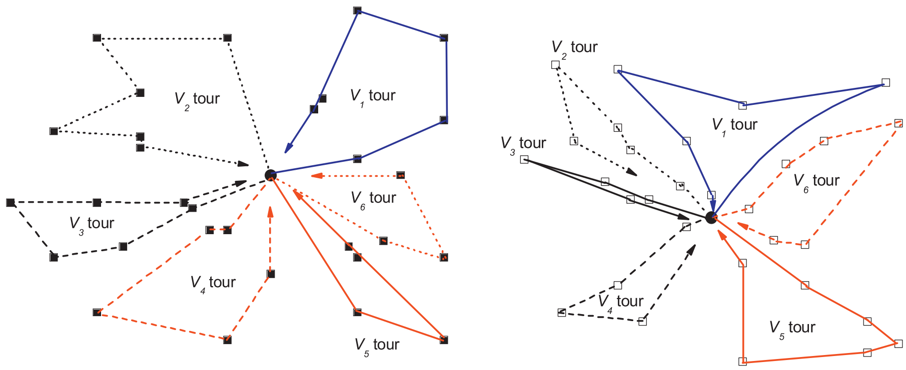

Fig. 4. Minimum-routing cost solution for the instance 30R-6V-3RD-3SD of Example 1.

the cross-dock layout do not affect the optimal level of the primary target but significantly influence on the value of the secondary target. For some problems, slightly better solutions can be found by adopting $\Delta > 0$.

After finding the minimum P/D routing-cost solutions, the vehicle routes were fixed and the VRPCD-TS model was solved again to minimize the total distribution time. Table 5 reports the best solutions found using the total distribution time as the secondary target for instances of Example 1 involving 30 or more requests. From the results shown in Table 5, it can be concluded that the best time-solutions using the sequential scheme are found in a very short CPU time but proving the optimality is much harder.

As shown in the next Section, they are indeed very good solutions from the time viewpoint and computationally less expensive than the best ones found using the least total distribution time as the primary target. Graphical representations of the pickup and delivery tours for the instance 30R-6V-3RD-3SD of Example 1 are shown in Fig. 4. All tours present tear-drop shapes with no crossing of route legs.

Moreover, detailed computational results for the problem instance 30R-6V-3RD-3SD are described in Table 6. By analyzing the starting of the unloading operations, it is concluded that lines of waiting vehicles arise at two strip dock doors. For instance, vehicle V5 should wait for the unloading of V6 at the receiving door RD1. Similarly, vehicles V3 and V2 wait for service at the strip dock doors RD1 and RD2, respectively. The same situation occurs at the stack doors with vehicles V2 and V3 at SD1, and vehicle V5 at SD3. Interestingly, the total P/D distribution time is just equal to 1793.7 time units close to the value obtained when the total distribution time is the primary target.

Table 7 presents a complete report of the computational results for the instance 40R-8V-3RD-3SD of Example 1. Vehicle queues involving up to 3 trucks are formed at some R/S dock doors like RD2 and SD2. This produces some delays in the unloading/reloading of some vehicles at the cross-dock facility. For instance, vehicle V6 completes the pickup tour at time 89.9 but it starts the unloading operations on the receiving door RD1 at time $t = 121.9$. This is so because V6 is preceded by V3 at RD1. A similar situation arises at the stack dock doors with the vehicles (V5, V2, V1, V3 and V7). Despite the total P/D distribution time is a secondary target, a very good solution featuring a total time of 2486.8 units is obtained. Similarly to the instance 30R-6V-3RD-3SD, pickup and delivery tours have tear-drop shapes and no edge crossing at all (see Fig. 5).

6.3. Minimizing the total distribution time as the primary target

The proposed VRPCD-TS formulation is again applied to a series of 18 problem instances of Example 1 with 10–40 requests but now using the least total P/D distribution time as the primary problem goal. The best solutions found, the computational times and the related optimality gaps, if any, are all shown in Table 8.

It can be observed that the sweeping-based formulation is able to solve problem instances with up to 20 requests, 4 vehicles and 3 strip/stack dock doors to optimality in a reasonable CPU time. For larger examples, there is a finite optimality gap after a CPU time of 3600 s that increases with the number of requests. Clearly, the optimality gap is much smaller when the problem goal is the minimum P/D vehicle routing cost. This is because such a target does not account for the costs associated to the movement of goods through the cross-dock facility. Nonetheless, the total distribution time for larger examples found through a sequential scheme is rather close to the best value obtained when using such a target as the primary objective function. Moreover, they are found at much lesser computational time. It is worth noting that the optimization of P/D routes and the cargo movement on the cross-dock are all separately NP-hard problems. The truck scheduling constraints defining the times at which to unload and load the vehicles permit to coordinate the three problems in the time domain, thus yielding an extremely difficult integrated problem. From Table 8, it is clear the strong influence of the cross-dock layout on the distribution time by comparing

Table 5
Best solutions for some instances of Example 1 with the distribution time as a secondary target.

|N|	|V|	|RD|	|SD|	Best solution (time units)	Gap (%)	CPU time (s)	
						To find the best solution	To prove optimality
30	6	3	3	1793.7	–	6.0	12.3
32	7	3	3	1925.8	2.28	109.0	3600*
34	7	3	3	2090.6	4.65	42.0	3600*
36	7	3	3	2147.6	–	216.0	1480.0
38	8	3	3	2264.3	1.63	137.0	3600*
40	8	3	3	2486.8	3.09	24.0	3600*
42	11	5	5	3026.5	–	88.2	3017.8
44	12	5	5	3081.9	–	1188.7	3441.0
46	12	4	4	3238.8	–	127.4	3532.5

* CPU time limit.

Table 6
Minimum-cost solution for the instance 30R-6V-3RD-3SD of Example 1.

Pick-up routes

Vehicle	Tour	Load collected	Vehicle returning time	Tour cost
V1	r30-r3-r9-r20-r1-r26	71	97.8	80.6
V2	r10-r19-r7-r8-r11-r18	71	108.9	91.7
V3	r28-r29-r16-r17-r5-r6	68	88.6	72.0
V4	r13-r25-r14-r15-r2	66	95.5	79.8
V5	r22-r23-r21-r24	70	90.2	74.2
V6	r27-r4-r12	60	69.6	56.1

Unloading operations

Receiving dock door	Vehicle	Service start time	Drop-off requests	Vehicle leaving time
RD1	V6	69.6	r27-r4-r12	100.1
	V5	100.1	r22-r23-r21-r24	135.6
	V3	135.6	r28-r16-r17-r5-r6	164.6
RD2	V4	95.5	r13-r25-r14-r15-r2	129.0
	V2	129.0	r10-r19-r7-r8-r11-r18	165.0
RD3	V1	97.8	r30-r3-r9-r20-r1-r26	133.8

Shipping operations

Shipping dock door	Vehicle	Vehicle arrival time	Ship-on requests	Service start-time	Service completion time
SD1	V4	133.0	r27-r1-r9-r3	133.8	164.7
	V3	166.6	r25-r23-r26	166.6	193.1
	V2	169.0	r17-r24-r16-r22-r12	193.1	228.6
SD2	V6	104.1	r5-r14-r6-r20-r30-r7	165.0	201.0
SD3	V1	135.8	r13-r15-r21-r2-r4	135.8	170.2
	V5	143.6	r28-r11-r10-r19-r18-r8	170.2	207.2

Delivery routes

Vehicle	Load to deliver	Vehicle departure time	Tour	Vehicle returning time	Tour cost
V1	68	170.2	r13-r15-r21-r2-r4	309.6	123.2
V2	70	228.6	r17-r24-r16-r22-r12	332.6	87.6
V3	63	193.1	r25-r29-r23-r26	273.3	65.6
V4	68	164.7	r27-r1-r9-r3	249.9	71.0
V5	73	207.2	r28-r11-r10-r19-r18-r8	324.5	99.8
V6	71	201.0	r5-r14-r6-r20-r30-r7	303.8	85.6
Total P/D vehicle routing cost					987.2
Total P/D vehicle usage time				1793.7	

the optimal values found for the instances 18R-4V-1RD-1SD and 18R-4V-2RD-2SD of Example 1. The total vehicle usage drops from 1031.5 to 965.7 by increasing the number of R/S dock doors by one.

The best solutions found after several CPU time milestones for problem instances that cannot be solved to optimality within the 3600 s CPU limit are reported in Table 9. Note that the minimum-time solutions found after the 900 s and 1800 s CPU milestones do not substantially differ from the best ones discovered in 3600 s. In other words, the proposed formulation finds good solutions in quite acceptable CPU times most of which is consumed in just raising the lower bound on the objective value to prove optimality. Therefore, high-quality solutions for large examples are usually discovered by the hybrid formulation in much shorter CPU times.

6.4. Solving VRPCS-TS problem instances with time windows

To test the hybrid formulation on problems with time windows, the same instances of Example 1 considered in the two previous Sections are again solved but now imposing time window constraints for the start of the service at P/D locations. The service time windows for the P/D sites of the requests are listed in Table B.3. For any P/D node, we choose a window width equal to 60

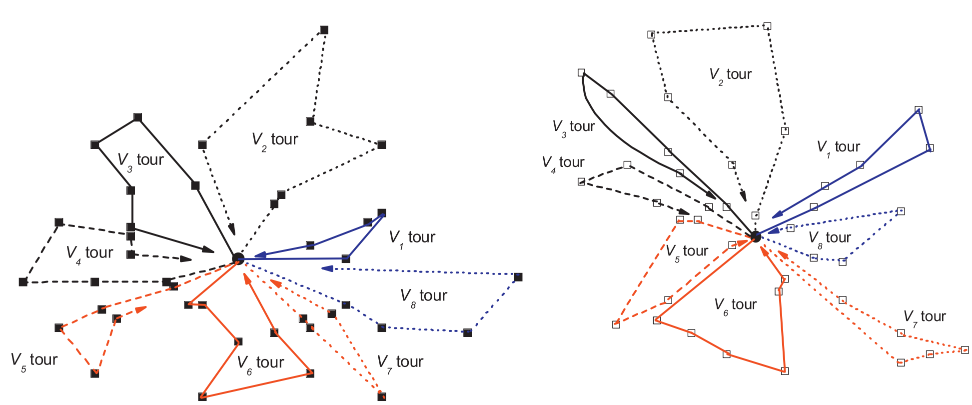

Fig. 5. Minimum cost solution for the instance 40R-8V-3RD-3SD of Example 1.

Table 7
Minimum cost solution for the instance 40R-8V-3RD-3SD of Example 1.

Pick-up routes

Vehicle	Tour	Load Collected	Vehicle returning time	Tour cost
V1	r26-r1-r9-r20-r38-r10	72	137.4	120.0
V2	r33-r39-r19-r7-r11	72	95.2	78.3
V3	r34-r18-r8-r17-r5-r6	64	90.4	74.6
V4	r28-r37-r16-r14-r29	74	91.5	74.1
V5	r25-r13-r2-r15-r22-r32	73	97.1	79.5
V6	r24-r21-r23-r27	74	89.9	73.1
V7	r35-r4-r36-r40	57	103.6	90.2
V8	r30-r31-r3-r12	54	61.7	48.9

Unloading operations

Receiving dock door	Vehicle	Service start time	Drop-off requests	Vehicle leaving time
RD1	V3	90.4	r34-r18-r8- r5-r6	121.9
	V6	121.9	r24-r21-r23-r27	159.4
RD2	V4	91.5	r28-r37-r16-r14	123.5
	V5	123.5	r13-r2-r15-r22-r32	156.5
	V1	156.5	r26-r1-r9-r38-r10	188.5
RD3	V8	61.7	r31-r3-r12	85.2
	V2	95.2	r33-r39-r19-r7-r11	131.7
	V7	131.7	r35-r4-r36-r40	160.7

Shipping operations

Shipping dock door	Vehicle	Vehicle arrival time	Ship-on requests	Service start-time	Service completion time
SD1	V6	161.4	r31-r3-r40-r18-r34-r8	161.4	198.9
	V5	160.5	r25-r26-r9-r1-r27	198.9	230.9
SD2	V4	125.5	r32-r12-r23	159.4	188.4
	V2	137.7	r4-r15-r36-r38-r21-r2	188.4	219.4
	V1	190.5	r14-r6-r13- r35	219.4	249.9
SD3	V8	87.2	r7-r33-r5	131.7	156.2
	V3	129.9	r24-r39-r22-r16	159.4	194.9
	V7	162.7	r19-r10-r37-r11-r28	194.9	229.9

Delivery routes

Vehicle	Load to deliver	Vehicle departure time	Tour	Vehicle arrival time	Tour cost
V1	69	249.9	r14-r6-r13-r20-r35	353.3	87.0
V2	61	219.4	r4-r15-r36-r38-r21-r2	357.7	123.1
V3	72	194.9	r24-r39-r22-r16-r17	310.3	98.5
V4	68	188.4	r32-r12-r23-r29	273.5	69.5
V5	71	230.9	r25-r26-r9-r1-r27	321.0	73.4
V6	74	198.9	r31-r3-r40-r18-r34-r8	301.6	84.9
V7	69	229.9	r19-r10-r37-r11-r28	342.4	96.2
V8	56	156.2	r7-r30-r33-r5	227.0	57.6
Total P/D vehicle routing cost					1329.1
Total P/D vehicle usage time				2486.8	

Table 8
Best time solutions for instances of Example 1 with 10–40 requests.

|N|	|V|	|RD|	|SD|	Best solution	Gap (%)	CPU (s)
10	2	1	1	475.6	–	16.2
12	3	1	1	577.8	–	25.8
14	3	1	1	799.9	–	13.4
16	4	1	1	934.3	–	122.5
		2	2	915.3	–	238.1
18	4	1	1	1031.5	–	154.8
		2	2	965.7	–	490.0
20	4	3	3	1202.7	–	2090.5
22	5	3	3	1324.1	5.96	3600[a]
24	5	3	3	1531.1	1.66	3600[a]
26	6	3	3	1573.6	8.88	3600[a]
28	6	3	3	1671.6	9.82	3600[a]
30	6	3	3	1738.4	9.74	3600[a]
32	7	3	3	1825.7	15.4	3600[a]
34	7	3	3	1959.9	14.6	3600[a]
36	7	3	3	2129.1	13.1	3600[a]
38	8	3	3	2396.2	25.9	3600[a]
40	8	3	3	2500.9	15.1	3600[a]

[a] CPU time limit.

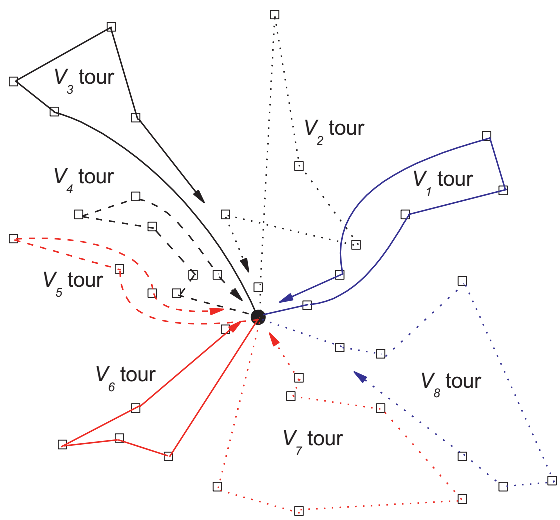

Fig. 6. Minimum-cost solution for the instance 40R-8V-3RD-3SD-TW of Example 1.

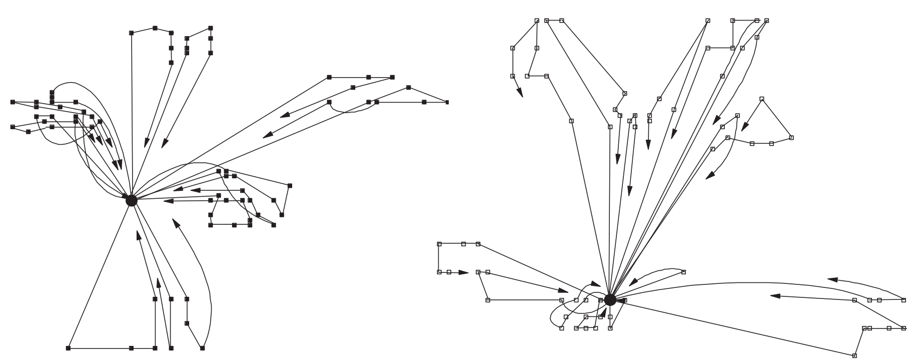

Fig. 7. Minimum-cost solution for the instance 70R-16V-7RD-7SD of Example 2.

time units. Usually, time windows distort the shape of the vehicle routes and cause some route overlapping at any feasible solution. Then, the variable Δ is allowed to take a finite value to let route overlapping. The extent of the route overlapping can be controlled by setting an upper bound on the value of Δ. When route overlapping is not allowed, such a bound is fixed at zero. Otherwise, a finite bound usually within the interval [0,0.5] is adopted and the variable Δ takes the best value inside that range. Although

Table 9
Evolution of the best time-solutions for large instances of Example 1 with the CPU time.

Example	CPU = 900 s		CPU = 1800 s		CPU = 3600 s	
	Best solution	Relative gap (%)	Best solution	Relative gap (%)	Best solution	Relative gap (%)
	Minimum time solutions					
30R-6V-3RD-3SD	1804.9	17.4	1758.5	11.8	1738.4	9.7
32R-7V-3RD-3SD	2016.8	36.2	1910.9	20.9	1825.7	15.4
34R-7V-3RD-3SD	2066.7	21.2	2066.7	21.2	1959.9	14.6
36R-7V-3RD-3SD	2129.1	13.8	2129.1	13.4	2129.1	13.1
38R-8V-3RD-3SD	No int.sol.	–	2642.8	47.8	2396.2	25.9
40R-8V-3RD-3SD	2558.4	20.5	2550.0	19.8	2500.9	15.1

Table 10
Minimum cost solutions for several instances of Example 1 with time-windows.

|N|	|V|	|RG|	|SG|	Primary objective (cost)				Secondary objective (time)		
				Objective function	Gap at 3600 s (%)	Time to find it (s)	CPU time (s)	Objective function	Gap (%)	CPU time (s)
28	6	3	3	967.7	8.26	2861	3600	2001.8	–	0.3
30	6	3	3	984.8	13.9	367	3600	2024.2	–	0.5
32	6	3	3	1039.0	1.52	2504	3600	2016.6	–	4.4
34	7	3	3	1088.1	27.4	1835	3600	2329.8	–	5.3
36	7	3	3	1233.4	11.1	525	3600	2421.2	–	12.8
38	8	3	3	1398.8	58.5	2598	3600	2721.4	–	152.3
40	8	3	3	1461.2	11.5	4859	5000	2751.4	–	14.0

Table 11
Minimum-cost solution for the instance 40R-8V-3RD-3SD-TW of Example 1.

Pick-up stage

Vehicle	Tour	Load Collected	Vehicle returning time	Tour cost
V1	r1-r26-r20-r38-r10	56	125.7	112.0
V2	r33-r11-r7-r19-r39	72	108.4	91.5
V3	r28-r6-r34-r5-r17-r8-r18	71	108.7	84.2
V4	r25-r29-r14-r16-r37	75	99.5	75.4
V5	r2-r15-r22-r32-r13	65	96.3	79.9
V6	r24-r27-r21-r23	74	97.5	78.3
V7	r35-r40-r36-r4	57	109.3	93.7
V8	r30-r9-r3-r31-r12	70	92.5	76.0

Unloading operations

Receiving dock door	Vehicle	Service start time	Drop-off requests	Vehicle leaving time
RD1	V8	92.5	r9-r3-r31-r12	124.0
	V5	124.0	r2-r15-r22-r32-r13	157.0
RD2	V4	99.5	r29-r14 -r37	124.0
	V1	125.7	r1-r26-r38-r10	149.7
	V7	149.7	r35-r36-r4	171.7
RD3	V6	97.5	r24-r21-r23	124.0
	V2	124.0	r33-r11-r7-r19-r39	160.5
	V3	160.5	r28-r6-r34-r5-r17-r8-r18	196.5

Shipping operations

Shipping dock door	Vehicle	Vehicle arrival time	Ship-on requests	Service start-time	Service completion time
SD1	V6	131.0	r3-r31-r9-r1	187.2	214.2
	V7	175.7	r18-r19-r28-r34-r8	214.2	243.2
SD2	V8	128.0	r7-r33-r37-r10-r11	171.8	205.8
	V4	126.0	r32-r24-r12-r17	207.7	229.2
	V3	202.5	r22-r39-r38-r21	229.2	257.6
SD3	V2	162.5	r4-r36-r15-r14-r2	171.9	204.4
	V5	165.0	r29-r23-r26	204.4	232.4
	V1	154.7	r5-r6-r13-r35	232.4	265.9

Delivery stage

Vehicle	Load to deliver	Departure time	Tour	Vehicle returning time	Tout cost
V1	75	265.9	r5-r6-r20-r13-r35	372.2	88.8
V2	64	204.4	r4-r36-r15-r14-r2	355.2	135.3
V3	56	257.6	r22-r39-r38-r21	386.7	115.8
V4	69	229.2	r25-r32-r24-r12-r16-r17	311.7	65.7
V5	55	232.4	r29-r23-r26	310.5	65.6
V6	75	214.2	r3-r31-r9-r1-r27	303.4	71.6
V7	71	243.2	r40-r18-r19-r28-r34-r8	374.1	113.6
V8	75	205.8	r7-r30-r33-r37-r10-r11	337.6	113.8
Total routing cost					1461.2
Total distribution time				2751.4	

counterintuitive, the TW-constrained problems are sometimes more difficult to solve than unconstrained problems because the nonzero overlapping Δ implies the enlargement of the solution space. In some cases, it is better to assume soft time windows and simultaneously incorporate an additional term in the objective function to penalize time window violations. Then, a pair of new continuous variables is needed for each request to measure the TW-violations. This solution scheme was used just to solve the instance 40R-8V-3RD-3SD with time windows.

Table 10 shows the best routing-cost solutions found for VRPCD-TS problem instances of Example 1 involving from 28 to 40 requests, 6-to-8 vehicles and 3 R/S dock doors. Moreover, the best time solutions discovered using a sequential scheme are also reported. Note that the search for the minimum routing-cost solution in the presence of time-window constraints is computationally more costly. In contrast, the sequential strategy using the least distribution time as a secondary target provides good time-solutions in a very short CPU time.

Table 12
Best routing cost solutions for several instances of Example 2.

Examples	Best solution found[a]		Relative gap	Binary vars	Cont. var	Constraints
	Total travel cost	Total distribution time				
50R-13V-6RD-6SD	1582.4	3697.8	0.044	2460	12,628	928,527
55R-13V-6RD-6SD	1637.4	3818.3	0.066	2701	14,042	1,037,399
60R-14V-7RD-7SD	1749.0	4059.0	0.076	3204	17,455	1,473,411
65R-15V-7RD-7SD	1971.7	4460.0	0.155	3737	21,139	2,045,390
70R-16V-7RD-7SD	2005.8	4201.2	0.096	4281	25,338	2,785,536

[a] After a CPU time limit of 5000 s.

Sketches of the vehicle tours and detailed computational results for the instance 40R-8V-3RD-3SD of Example 1 with time windows are shown in Fig. 6 and Table 11, respectively. The starting times of unloading/reloading operations at P/D nodes and the compliance with the time windows are depicted in Table B.3 (Appendix B).

6.5. Solving several instances of the larger Example 2

Example 2 is a larger VRPCD-TS case study involving up to 70 transportation requests, 16 P/D vehicles and a cross-dock facility with 7 strip/stack dock doors. The vehicle capacity is equal to 90 units. Data for Example 2 are given in Tables C.1 and C.2 of Appendix C. Five problem instances of Example 2 were generated by considering the first N entries of Table C.1 with N = 50, 55, 60, 65 and 70. They have been solved using the least total routing cost as the primary problem target and a CPU time limit of 5000 s. Moreover, the minimum distribution time was adopted as a secondary target to be achieved in a sequential manner. Computational results for the problem instances 50R-13V-6RD-6SD, 55R-13V-6RD-6SD, 60R-14V-7RD-7SD, 65R-15V-7RD-7SD and 70R-16V-7RD-7SD are shown in Table 12. They include the model size, the total routing cost and the total distribution time at the best solutions and the related relative gap for each problem instance. None of the instances was solved to optimality within the CPU time limit. However, good solutions were already discovered in CPU times below 1500 s. A graphical representation of the best solution for the instance 70R-16V-7RD-7SD of Example 2 is shown in Fig. 7. More detailed information on the best solution of the instance 70R-16V-7RD-7SD is given in Tables C.3 and C.4 of Appendix C.

7. Conclusions

This work presents an MILP monolithic formulation for the scheduling of single cross-dock distribution systems that accounts for vehicle routing, dock door assignment and truck scheduling. The model considers a two-stage cross-dock facility where the incoming freights are received at strip dock doors, stored temporarily on the dock for screening and sorting the goods by destination and moved them to the stack door area to reload into outbound trucks. The cross-dock is assumed to have multiple strip/stack dock doors but the number of them may be lower than the number of P/D trucks, e.g. the dock doors are scarce resources. Then, queues of trucks can be formed in front of every dock door. The ordering of them on the vehicle lines and the timing of the unloading and reloading operations at the dock doors are selected by the model. Dock doors are designated as either strip or stack doors but not both. Though the approach assumes that the same homogeneous vehicle fleet accomplishes the pickup and delivery tasks, it can still be applied if a truck is either inbound or outbound by ignoring the pair of constraints (25) and (28). When some loads are collected and delivered by the same truck, the formulation assumes that they are not unloaded at the cross-dock and remain inside the vehicle. Routes and schedules for pickup/delivery vehicles are carefully chosen so as to get a better coordination with the timing of unloading and reloading operations at the cross-dock. Moreover, inbound vehicles can return to the cross-dock at different times and each order must be collected and delivered by a single vehicle. Besides, the freights to be delivered do not usually arrive at the cross-dock in the order required by the loading sequence of an outbound vehicle. Then, the loading of a delivery truck can start only if all the assigned goods are available in front of the stack door. As the content of the temporary storage is traced by the model, limitations on the inventory capacity could be handled.

Due to the detailed description of cross-docking issues, the proposed formulation is rather complex. To speed up its resolution, some valid inequalities aimed at pruning the space of feasible solutions were included. Moreover, additional restrictions for (a) avoiding symmetric solutions and (b) mimicking the well known VRP sweep-heuristic algorithm were embedded into the model. Although the resulting hybrid approach can no longer guarantee optimality, it is computationally much more efficient.

The hybrid formulation was applied to a wide variety of problem instances of two medium-size examples with the larger one involving up to 70 requests, 16 vehicles and 7 strip/stack doors. The least vehicle routing cost and the minimum distribution time were alternatively chosen as the problem targets. The VRPCD-TS approach was first validated against the exact approach by solving rather small examples and comparing their results and computational requirements. Afterwards, larger examples were tackled. When the total routing cost was minimized, almost all the problem instances were solved to optimality in acceptable CPU times. The optimality gap always remains below 1.6% for all instances of Example 1 within the time limit of 3600 s. Even for the larger instances of Example 2, good solutions are usually discovered at acceptable CPU times. By fixing the optimal routes and solving again the hybrid representation with the minimum distribution time as a secondary target, very good solutions from the time viewpoint were discovered. They are found in very short CPU times but proving their optimality is much harder. Clearly, the least total distribution time as a primary target is computationally less efficient. Nonetheless, the best solutions are often identified at reasonable CPU times but the average optimality gap for the larger examples rises to 15.6% after 3600 s. At the best solutions, freight stays on the temporary storage during some period of time waiting (a) for the arrival of the other goods to be delivered by the vehicle, and (b) for the turn time of the delivery truck staying on the line of the assigned stack door. In addition, the proposed approach was also applied to solve problem instances with specific time windows within which the service of pickup and delivery nodes should be started. Time-window constrained cross-docking problems are sometimes more difficult to solve because the routes are distorted and no longer look like tear drops. Then, a non-zero overlapping parameter Δ is to be adopted to get feasible solutions and consequently a larger solution space is to be explored. Future work will be focused on the scheduling of cross-dock systems transporting a number of different products between suppliers to customers using a heterogeneous fleet of vehicles.

Acknowledgment

The authors acknowledge financial support from FONCYT-ANPCyT under Grant PICT 2010-1073, from CONICET under Grant PIP-2221, and from Universidad Nacional del Litoral under CAI+D 66-335.

Appendix A. Mimicking the sweeping algorithm

In order to mimic the sweeping algorithm, the following set of constraints has been added to the problem formulation.

Angular limits and width of the vth-circular sector. As stated by Eq. (A1), the upper angular limit of sector v is the lower limit of sector ($v+1$). Moreover, the set of zones defined by the model should cover the whole region to be served. By Eq. (A2), the sum of their angular widths must be equal to 2π.

$$\phi^P_{v+1} = \phi^P_v + \Delta\phi^P_v \quad \forall v \in V(v < \left|V\right|) \tag{A1}$$

$$\sum_{v \in V} \Delta\phi^P_v = 2\pi \tag{A2}$$

Unused sectors arising first in the set V. A number of angular zones equal to the number of available vehicles should be predefined but some zones could be fictitious because not all the vehicles might be used. The binary variable U_v^P has a zero value for a fictitious zone. The constraint (A3) drives the angular width of any fictitious sector to zero. On the other hand, Eq. (A4) ensures that fictitious sectors, if any, will arise first.

$$\Delta\phi_v^P \le 2\pi U_v^P \quad \forall v \in V \tag{A3}$$

$$U_{v+1}^P \ge U_v^P \quad \forall v \in V(v < |V|) \tag{A4}$$

Allocating nodes to vehicles. Through Eq. (1) each pickup location must be assigned to exactly one vehicle. If vehicle v is not used ($U_v^P = 0$), then Eq. (A5) does not allow to assign customer locations to this vehicle.

$$YP_{r,v} \le U_v^P \quad \forall r \in R, v \in V \tag{A5}$$

Feasible allocation of nodes to the circular sector v. For every zone before the last one, all pickup locations featuring an angular coordinate θ_r^P within the sector v, i.e. $\theta_r^P \in [\phi_v^P, \phi_{v+1}^P]$ must be allocated to vehicle v. This condition is enforced by Eqs. (A6) and (A7). The vehicle assignment for locations just on the boundary between sectors v and $v+1$ is left to the model.

$$\left\{\begin{array}{l} \phi_v^P \le \theta_r^P + 2\pi\,(1 - YP_{rv}) \\ \phi_{v+1}^P + \Delta \ge \theta_r^P\, YP_{r,v} \end{array}\right\} \quad \forall r \in R, v \in V\,(v < |V|) \qquad \begin{array}{r} (A6) \\ (A7) \end{array}$$

The tuning parameter Δ allows an overlap of magnitude Δ between two adjacent sectors and it is used in time-windows constrained problems. In that case, locations within the Δ-sized overlapped area can be allocated to the sector v or $v+1$.

Allowing the first used angular sector to start at the best angular location. The last zone requires a special constraint because the rotating ray may start its movement from an initial polar angle ϕ_1^P larger than (min θ_r^P). The pickup locations with an angular coordinate $\theta_r^P \in [0, \phi_1^P)$ must be allocated to the last sector $v = |V|$. By defining the continuous variable ξ_r^P taking the value 1 whenever the pickup location of request r satisfies the condition: $\theta_r^P \in \left[0, \phi_1^P\right)$, Eqs. (A8) and (A9) assign request r to the last sector.

$$YP_{r,v} \ge \xi_r^P \quad \forall r \in R, v = |V| \tag{A8}$$

Eq. (A9) reduces to: $\theta_r^P \xi_r^P \le \phi_v^P, v \in V$ for every existent sector v. If $\xi_r^P = 1$, then Eq. (A8) becomes: $\theta_r^P \le \phi_v^P$. For fictitious sectors, the constraint (A9) becomes redundant.

$$\theta_r^P(\xi_r^P + U_v^P - 1) \le \phi_v^P \quad \forall r \in R, v \in V \tag{A9}$$

Eq. (A10) is incorporated into the problem formulation to speed-up the convergence rate.

$$\theta_r^P \ge \phi_v^P - 2\pi\xi_r^P \quad \forall r \in R, v = 1 \tag{A10}$$

Besides, Eq. (A11) replaces Eq. (A6) for the last sector. This constraint forcing a request assigned to the last sector $v = |V|$ to have

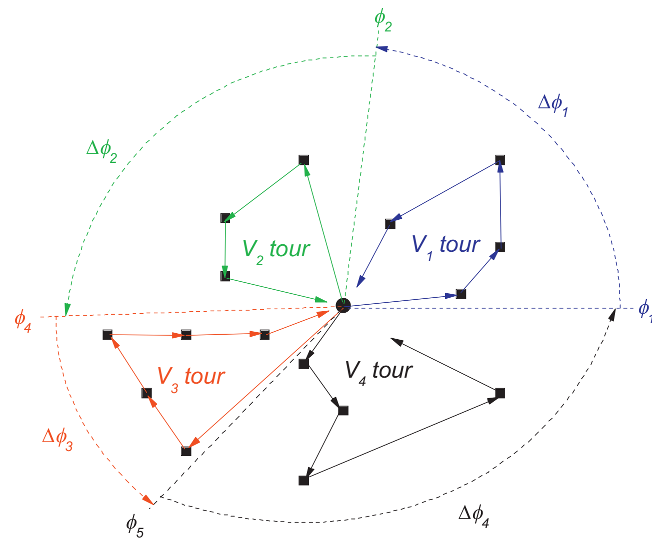

Fig. A1. Illustrating the angular sectors defined by the sweeping.

an angular coordinate $\theta_r^P \ge \phi_{|V|}^P$ no longer applies if ξ_r^P is equal one. When $\xi_r^P = 0$, Eq. (A11) looks similar to Eq. (A6).

$$\phi_v^P \le \theta_r^P + 2\pi(1 + \xi_r^P - YP_{rv}) \quad \forall r \in R, v = |V| \tag{A11}$$

The proposed set of constraints is just written for pickup routes but an identical set should be proposed for the delivery tours. The constraints (A1)–(A11) eliminate symmetrical solutions related to the assignment of vehicles to tours by allocating vehicles in a growing angular order. So, the vehicle *v1* will be allocated to the first angular sector (whose lower angular border have the minimum value), the vehicle *v2* to the next region (whose lower angular limit is the upper angular limit of the sector visited by the vehicle *v1*) and so on. It is worth noting that the constraints (A1)–(A11) not only mimic but also improve the sweep-heuristic algorithm because the width of the angular areas is adjusted in order to further reduce the objective function value. In addition, the best starting polar angle (i.e., the initial ray position) is optimized. Fig. A1 illustrate the adjustment of parameters φ_v and $\Delta\varphi_v$ for a fleet $V = \{v1, v2, v3, v4\}$.

Appendix B. Problem data and computational results for Example 1 constraints

Tables B.1–B.3.

.

Table B.1
Data for the 46 transportation requests of Example 1.

Request	Load	X coord	Y coord	Time Windows		X coord	Y coord	Time window	
				a	b			a	b
		Pick-up stage				Delivery stage			
r1	10	41	49	10	70	20	20	260	320
r2	7	35	17	10	70	31	52	280	340
r3	13	55	45	10	70	24	12	210	270
r4	19	55	20	20	80	35	40	200	260
r5	26	15	30	40	100	41	37	220	280
r6	3	25	30	20	80	53	52	260	320
r7	5	20	50	40	100	45	30	210	270
r8	9	10	43	20	80	40	25	320	380
r9	16	55	60	30	90	11	14	260	320
r10	16	30	60	40	100	65	7	280	340

Table B.1 (*Continued*)

Request	Load	X coord	Y coord	Time Windows		X coord	Y coord	Time window	
				a	b			a	b
		Pick-up stage				Delivery stage			
r11	12	20	42	20	80	60	12	280	340
r12	19	50	35	40	100	13	52	220	280
r13	23	30	25	20	80	63	65	300	360
r14	20	15	10	40	100	47	47	260	320
r15	8	30	5	0	60	40	60	280	340
r16	19	10	20	20	80	20	55	280	340
r17	2	5	30	20	80	30	42	300	360
r18	12	20	40	30	90	40	3	290	350
r19	17	15	60	40	100	60	5	300	360
r20	9	45	65	30	90	65	56	260	320
r21	11	45	20	0	60	20	68	290	350
r22	18	45	10	40	100	10	69	260	320
r23	29	55	5	40	100	5	48	240	300
r24	12	44	22	10	70	22	50	220	280
r25	8	28	25	0	60	25	39	240	300
r26	15	40	47	20	80	22	39	280	340
r27	22	48	23	0	60	31	33	280	340
r28	7	26	29	10	60	50	20	300	360
r29	11	18	22	20	80	18	43	240	300
r30	8	45	38	10	70	50	29	210	270
r31	14	53	43	30	90	18	15	240	300
r32	9	40	19	20	80	27	42	220	280
r33	17	29	51	0	60	60	41	240	300
r34	12	20	36	20	80	39	22	300	360
r35	14	50	25	10	70	45	42	320	380
r36	10	67	19	20	80	37	85	270	330
r37	17	16	24	20	80	71	8	280	340
r38	6	47	85	10	70	17	83	280	340
r39	21	21	66	60	120	5	74	280	340
r40	14	74	31	0	60	30	7	220	280
r41	19	8	70	–	–	66	58	–	–
r42	11	47	47	–	–	18	37	–	–
r43	20	29	25	–	–	9	5	–	–
r44	13	75	15	–	–	55	28	–	–
r45	9	12	73	–	–	31	69	–	–
r46	26	32	2	–	–	67	11	–	–

Cross-dock Cartesian coordinates: $X_w = 35$, $Y_w = 35$

Table B.2
Vehicle transfer times between strip and stack dock doors for Example 1.

	SD1	SD2	SD3
RD1	2	4	8
RD2	4	2	5
RD3	7	6	2

Table B.3
Service start times at P/D locations at the best solution for the instance 40R-8V-3RD-3SD-TW of Example 1.

Vehicle	Pickup routes			Delivery routes		
	Request	Service start time	Time window	Request	Service start time	Time window
V1	r1	15.2	10–70	r5	272.3	220–280
	r26	20.0	20–80	r6	297.2	260–320
	r20	42.2	30–90	r20	310.9	260–320
	r38	64.6	10–70	r13	322.4	300–360
	r10	96.5	40–100	r35	356.7	320–380
V2	r33	17.1	0–60	r4	209.4	200–260
	r11	33.7	20–80	r36	270.0	270–330
	r7	44.6	40–100	r15	297.7	280–340
	r19	57.3	40–100	r14	314.6	260–320
	r39	69.7	60–120	r2	335.8	280–340
V3	r28	10.8	10–60	r22	299.8	260–320
	r6	20.0	20–80	r39	311.0	280–340
	r34	28.9	20–80	r38	330.7	280–340
	r5	46.7	40–100	r21	347.7	290–350
	r17	62.4	20–80			
	r8	77.3	20–80			
	r18	90.0	30–90			

Table B.3 (*Continued*)

Vehicle	Pickup routes			Delivery routes		
	Request	Service start time	Time window	Request	Service start time	Time window
V4	r25	12.2	0–60	r25	240.0	240–300
	r29	24.7	20–80	r32	245.7	220–280
	r14	40.0	40–100	r24	257.4	220–280
	r16	55.7	20–80	r12	269.6	220–280
	r37	73.6	20–80	r16	281.5	280–340
				r17	302.2	300–360
V5	r2	18.0	10–70	r29	251.2	240–300
	r15	32.9	0–60	r23	267.8	240–300
	r22	50.8	40–100	r26	293.4	280–340
	r32	65.2	20–80			
	r13	80.0	20–80			
V6	r24	15.8	10–70	r3	239.7	210–270
	r27	22.8	0–60	r31	249.5	240–300
	r21	34.4	0–60	r9	260.0	260–320
	r23	55.1	40–100	r1	274.5	260–320
				r27	294.0	280–340
V7	r35	18.1	10–70	r40	271.7	220–280
	r40	46.1	0–60	r18	290.0	290–350
	r36	65.5	20–80	r19	313.0	300–360
	r4	80.0	20–80	r28	334.9	300–360
				r34	348.0	300–360
				r8	360.6	320–380
V8	r30	10.4	10–70	r7	217.0	210–270
	r9	36.7	30–90	r30	223.6	210–270
	r3	55.4	10–70	r33	241.3	240–300
	r31	61.3	30–90	r37	280.0	280–340
	r12	73.3	40–100	r10	290.0	280–340
				r11	300.6	280–340

Appendix C. Data and computational results for Example 2

Tables C.1–C.4

Table C.1
Data for the 70 transportation orders of Example 2.

Request	Load	Pickup routes		Delivery routes		Request	Load	Pickup routes		Delivery routes	
		Xcoord	*Ycoord*	*Xcoord*	*Ycoord*			*Xcoord*	*Ycoord*	*Xcoord*	*Ycoord*
1	10	42	15	10	35	36	20	65	55	20	85
2	20	70	58	88	30	37	20	25	55	90	35
3	10	29	53	66	71	38	20	40	63	27	67
4	20	75	55	33	32	39	10	60	60	65	85
5	20	40	66	60	80	40	20	28	52	92	30
6	10	50	30	30	30	41	10	42	68	0	45
7	10	22	49	68	63	42	20	44	5	35	30
8	10	38	15	65	82	43	10	53	30	20	80
9	10	63	58	95	30	44	10	66	55	50	40
10	10	45	68	10	40	45	40	25	52	5	45
11	20	25	57	43	68	46	40	40	15	62	80
12	20	49	37	61	68	47	20	45	65	85	35
13	10	28	55	28	30	48	10	48	30	22	85
14	30	68	60	28	35	49	10	42	65	2	40
15	10	20	55	25	85	50	20	53	35	55	80
16	30	45	70	67	85	51	30	31	51	58	66
17	10	48	40	26	32	52	10	72	55	30	32
18	10	30	50	35	66	53	15	23	54	40	66
19	10	60	55	18	75	54	12	47	41	56	62
20	30	65	60	88	35	55	18	28	51	40	68
21	20	24	51	40	66	56	8	45	32	64	63
22	20	30	52	0	40	57	10	47	35	15	75
23	10	42	66	8	40	58	20	23	55	58	75
24	10	49	35	36	69	59	15	46	41	45	71
25	40	42	10	35	32	60	5	27	5	38	35
26	30	40	5	95	35	61	14	51	32	48	69
27	10	25	50	25	35	62	22	54	32	37	68
28	10	47	40	87	30	63	10	23	52	22	75
29	10	55	38	43	67	64	12	46	36	59	64
30	30	38	5	33	35	65	20	50	35	30	35
31	10	20	50	60	85	66	15	50	31	72	64

Table C.1 (*Continued*)

Request	Load	Pickup routes		Delivery routes		Request	Load	Pickup routes		Delivery routes	
		Xcoord	*Ycoord*	*Xcoord*	*Ycoord*			*Xcoord*	*Ycoord*	*Xcoord*	*Ycoord*
32	10	45	30	8	45	67	10	38	70	15	80
33	20	35	5	25	30	68	6	26	54	38	72
34	10	45	35	32	30	69	20	40	69	55	85
35	10	35	69	85	25	70	25	25	56	39	67

(*) Cross-dock Cartesian coordinates: $X_w = 40$, $Y_w = 50$

Table C.2
Vehicle transfer times between strip and stack dock doors for Example 2.

Receiving doors	Shipping doors						
	SD1	SD2	SD3	SD4	SD5	SD6	SD7
RD1	2	4	8	2	1	3	5
RD2	4	2	5	3	6	1	4
RD3	7	6	2	1	3	5	2
RD4	8	5	4	1	4	6	3
RD5	6	9	5	4	2	1	4
RD6	8	6	6	4	2	2	1
RD7	7	7	5	4	4	3	2

Table C.3
Best routing cost solution for the instance 70R-16V-7RD-7SD of Example 2.

Pick-up routes

Vehicle	Tour	Load Collected	Vehicle returning time	Tour cost
V1	r44-r2-r4-r52-r36-r19	90	93.9	72.9
V2	r39-r20-r14-r9	80	78.1	60.1
V3	r23-r49-r41-r16-r10-r47	90	65.5	44.5
V4	r35-r67-r69-r5-r38	80	62.5	44.0
V5	r70-r37-r11-r13	75	52.8	35.8
V6	r3-r68-r58-r15-r53-r22	81	60.6	41.4
V7	IDLE			
V8	r40	20	28.8	24.3
V9	r45-r63-r51	80	51.8	34.3
V10	r55-r21-r31-r7-r27-r18	78	59.2	40.6
V11	r60-r33-r30-r8	65	117.9	102.9
V12	r46-r26	70	105.0	90.0
V13	r31-r1-r25-r42	80	109.5	91.5
V14	r64-r56-r48-r6-r66-r24-r57-r34	85	70.9	49.9
V15	r59-r43-r61-r65-r12	79	66.2	47.9
V16	r54-r29-r62-r50-r17-r28	84	69.3	49.5

Delivery routes

Vehicle	Load to deliver	Departure time	Tour	Vehicle returning time	Tout cost
V1	67	206.5	r54-r64-r56-r7-r66-r3	302.4	79.5
V2	50	146.1	r51-r12	212.5	55.3
V3	80	172.5	r46-r16-r8	278.5	88.5
V4	80	196.7	r58-r39-r31-r5-r50	306.8	91.6
V5	79	166.2	r61-r69-r59-r11-r29	261.7	77.2
V6	78	130.0	r70-r55-r21-r53	184.0	36.4
V7	68	145.5	r18-r48-r15-r68-r24-r62	246.5	84.4
V8	90	187.0	r38-r63-r19-r43-r36-r67-r37	301.1	92.6
V9	90	145.5	r32-r45-r41-r22-r49	252.7	86.7
V10	40	145.5	r27-r1-r23-r10	238.7	75.2
V11	80	211.5	r65-r17-r33-r14	279.8	50.3
V12	90	192.3	r30-r34-r6-r13-r52-r4	264.1	50.8
V13	65	148.5	r25-r42-r60	204.6	41.6
V14	10	147.0	r44	177.8	28.3
V15	90	232.5	r35-r28-r2-r40-r9-r47	377.1	123.5
V16	80	181.0	r20-r37-r26	312.9	114.3
Total routing cost					2005.8
Total distribution time				4201.2	

Table C.4
Cross-dock operations at the best solution for the instance 70R-16V-7RD-7SD.

Unloading operations

Receiving dock door	Vehicle	Service start time	Drop-off requests	Vehicle leaving time
RD1	V6	62.5	r3-r68-r58-r15-r22	96.0
	V1	96.0	r44-r2-r4-r52-r36-r19	141.5
RD2	V5	52.8	r70-r37-r13	80.8
	V14	80.8	r64-r56-r48-r6-r66-r24-r57-r34	123.8
RD3	V11	123.8	r60-r30-r8	146.8
	V10	59.2	r55-r21-r31-r7-r18	93.7
RD4	V16	93.7	r54-r29-r62-r50-r17-r28	136.2
	V9	51.8	r63-r51	72.3
	V3	72.3	r23-r49-r41-r10-r47	102.8
	V13	109.5	r31-r1-r42	120.0
RD5	V8	28.8	r40	39.3
	V15	66.2	r59-r43-r61-r65-r12	106.2
RD6	V2	78.1	r39-r20-r14-r9	118.6
RD7	V4	62.5	r35-r67-r69-r38	93.0
	V12	105.0	r46-r26	140.5

Shipping operations

Shipping dock door	Vehicle	Vehicle arrival time	Ship-on requests	Service start-time	Service completion time
SD1	V5	84.8	r61-r69-r59-r29	136.2	166.2
	V4	100.0	r58-r39-r31-r50	166.2	196.7
SD2	V8	48.3	r38-r63-r19-r43-r36-r67-r37	141.5	187.0
	V15	115.2	r35-r28-r2-r40-r9-r47	187.0	232.5
SD3	V16	138.2	r20-r37-r26	140.5	181.0
	V11	151.8	r65-r17-r14	181.0	211.5
SD4	V6	98.0	r70-r55-r21	98.0	130.0
	V10	94.7	r1-r23-r10	130.0	145.5
	V13	121.0	r60	145.5	148.5
SD5	V2	120.6	r51-r12	120.6	146.1
	V12	144.5	r30-r34-r6-r13-r52-r4	146.8	192.3
SD6	V14	124.8	r44	141.5	147.0
	V3	108.8	r46-r8	147.0	172.5
	V1	144.5	r54-r64-r56-r7-r66-r3	172.5	206.5
SD7	V9	75.3	r32-r41-r22-r49	120.0	145.5
	V7	–	r18-r48-r15-r68-r24-r62	145.5	180.0

References

Arabani, A. R. B., Ghomi, S. M. T. F., & Zandieh, M. (2011). Meta-heuristics implementation for scheduling of truck in a cross-docking system with temporary storage. *Expert Systems with Applications*, *38*, 1964–1979.

Bartholdi, J., & Gue, K. (2000). Reducing labor costs in an LTL crossdocking terminal. *Operations Research*, *48*(6), 823–832.

Boysen, N., & Fliedner, M. (2010). Cross dock scheduling: classification, literature review and research agenda. *Omega*, *38*(6), 413–422.

Chen, P., Guo, Y., Lim, A., & Rodrigues, B. (2006). Multiple crossdocks with inventory and time windows. *Computers and Operations Research*, *33*(1), 43–63.

Dondo, R., & Cerdá, J. (2013). A sweep-heuristic based formulation for the vehicle routing problem with cross-docking. *Computers and Chemical Engineering*, *48*, 293–311.

Gillet, B., & Miller, L. (1974). A heuristic algorithm for the vehicle dispatch problem. *Operations Research*, *22*, 340–349.

Lee, Y. H., Jung, J. W., & Lee, K. M. (2006). Vehicle routing scheduling for cross-docking in the supply chain. *Computers & Industrial Engineering*, *51*, 247–256.

Li, Z. P., Low, M. Y. H., Shakeri, M., & Lim, Y. G. (2009). Crossdocking planning and scheduling: Problems and algorithms. *SIMTech Technical Reports*, *10*(July–September (3)), 159–167.

Miao, Z., Yang, F., Fu, K., & Xu, D. (2012). Transshipment service through crossdocks with both soft and hard time windows. *Annals of Operations Research*, *192*(1), 21–47.

Tsui, L. Y., & Chang, C. H. (1992). An optimal solution to a dock door assignment problem. *Computers and Industrial Engineering*, *23*(1–4), 283–286.

Van Belle, J., Valckenaers, P., & Cattrysse, D. (2012). Cross-docking. State of the art. *Omega*, *40*, 827–845.

Wen, M., Larsen, J., Clausen, J., Cordeau, J. F., & Laporte, G. (2009). Vehicle routing with cross-docking. *Journal of the Operational Research Society*, *60*, 1708–1718.

Yu, W., & Egbelu, P. J. (2008). Scheduling of inbound and outbound trucks in cross docking systems with temporary storage. *European Journal of Operational Research*, *184*, 377–396.