Computers and Chemical Engineering 88 (2016) 170–184

Contents lists available at ScienceDirect

Computers and Chemical Engineering

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

Operational planning of forward and reverse logistic activities on multi-echelon supply-chain networks

Rodolfo G. Dondo*, Carlos A. Méndez

INTEC (Universidad Nacional del Litoral – CONICET), Güemes 3450, 3000 Santa Fe, Argentina

ARTICLE INFO

Article history:
Received 15 September 2015
Received in revised form 22 February 2016
Accepted 23 February 2016
Available online 2 March 2016

Keywords:
Forward and reverse logistics
Integrated supply chains
Multi-echelon networks
Columns generation
Decomposition

ABSTRACT

Distribution activities arising from supply-chains of chemical and food industries involve the shipping of products directly and/or via distribution-centers. Also, due to growing ecology concerns, the recycling of recoverable-materials is becoming a common practice. In this paper, a distribution and recovering problem has been studied and modeled. The solution to the problem-model computes the forward and backward flows on a supply-chain network of a company that take into account 'green logistics' considerations. In this problem, vehicles departing from plants/distribution-centers perform delivery of products and pick-up of recyclables at the lowest network-level. At a higher level, larger vehicles re-supply distribution-centers with products and bring back to plants recyclable goods. The operation must coordinate the vehicles-tours to assure efficient forward and backwards flows. The paper presents a column-generation based decomposition-approach for finding near-optimal solutions to the problem. We also present computational results on test problems derived from a real case-study.

1. Introduction

Due to the increasing pressure for reducing costs, inventories and the ecological footprint, and in order to remain competitive in the global marketplace, enterprise-wide optimization (EWO) has become a major goal of the chemical industry. EWO is concerned with the coordinated optimization of the operations of a supply chain, supply of material, manufacturing and distribution of products. The main objectives in EWO are maximization of profits, asset utilization, responsiveness to customers, minimization of costs, minimization of inventory levels and minimization of the ecological footprint (Grossmann, 2012). In addition, as the importance of product recovery has grown swiftly in the past few decades and the environmental concerns have led to an increase in recovery activities and in sustainability of supply chains, the 'green logistics' subject have received a growing research interest. Consumer's inclination toward green logistics, legal pressure and possible economic benefit are among the main reasons which led manufacturers to integrate recovery activities into their processes (Ilgin and Gupta, 2010). Distribution is concerned with shipment and storage of products downstream from the supplier side to the customer side in the supply chain and is a key issue for EWO. Typically products are manufactured in one or more factories, moved to distribution centers for intermediate storage and shipped to retailers or final consumers. On the other hand, reverse logistics is the process of recovering goods from consumers and retailers for the purpose of capturing value, recycling or a proper disposal. The growing ecology concerns and advances about green supply-chain management concepts make reverse logistics a relevant research issue. Reverse logistics may include the returning of containers with disposed materials, pallets with recyclable goods, equipment and machines. Normally, forward logistics deal with events that bring products from manufacturers toward customers but in reverse logistics, a resource goes back in the supply chain. For example, reusable packaging systems require a closed-loop logistics system. A distribution network generally includes factories at the higher network-level, distribution centers at the intermediate level, and retailers and end-consumers at the lowest level; with the products going from the highest to the lowest level and the reusable goods going upwards to higher levels. A distribution center can be regarded as an intermediate facility that allows the aggregation of products coming from different factories and destined to different retailers. Products can be temporarily stored in those centers before they are sent to the destinations and pallets with reusable goods may be temporarily stored there before going back to manufacturers and/or to recycling or disposal sites. Freight can be moved by trucks in two different modalities: a full truckload (FTL) mode or a less than truckload (LTL) mode. FTL is cheaper per transported unit and is usually employed to move a large quantity of products from plants toward distribution centers and to move recovered

* Corresponding author. Tel.: +54 342 4559175.
E-mail address: rdondo@santafe-conicet.gov.ar (R.G. Dondo).

http://dx.doi.org/10.1016/j.compchemeng.2016.02.017
0098-1354/© 2016 Published by Elsevier Ltd.

Nomenclature

Sets

A	arcs of the routes network
I	customers
P	plants
R^+	replenishment trips
R^-	recovery trips
R	pick-up and delivery routes
T	distribution centers

Parameters

a_{ir}	binary parameter denoting that site i is visited by pick-up and delivery route r
b_{pr}/b_{tr}	binary parameter denoting that route r starts/ends on plant p/distribution center t
$b_{tr}^{start}/b_{tr}^{start}$	binary parameter denoting that replenishment/recovery trip r starts on plant p/distribution center t
$b_{pr}^{end}/b_{tr}^{end}$	binary parameter denoting that replenishment/recovery trip r ends on plant p/distribution center t
cf_v	fixed cost of using a pick-up and delivery vehicle
c_r	cost of route r
c_{ij}	traveling cost for go from location i to location j
i_p^0/i_t^0	initial inventory of pallets with products on plant p/distribution center t
i_p/i_t	binary parameter indicating that route designed by the slave routes-generator problem is based on plant p/distribution center t
$i_{t'}$	binary parameter indicating that a vehicle depart from a distribution center after the arrival of a replenishment convoy
M_C, M_L, M_T	upper bounds for the travel cost (C), the transported load (L) and the travel time (T)
q	transport capacity of the pick-up-and-delivery vehicle
q_0	transport capacity of the vehicle traveling a replenishment/recovery trip
st_i	stop time at the pick-up/delivery site i
t_0	travel time of a replenishment convoy
$t^{\max}$	maximum allowed routing time
$t_i^{\min}$	earliest arrival time at customer i
$t_i^{\max}$	latest arrival time at customer i
x_p/x_t	binary parameter stating that plant p/distribution center t belongs to the route designed by a slave routes-generator problem
α_r	quantity of pallets with products to deliver by route r
β_r	quantity of pallets with goods to recover by route r
γ_i	quantity of pallets with products to deliver to customer i
ε_i	quantity of pallets with goods to recover from customer i
δ_p^f	demand of pallets with recovered goods by plant p
δ_p^o	quantity of pallets with products on distribution center t at the start of the planning horizon
π_i	price associated to customer i
π_p	price associated to balances of pallets with products on plant p
π_t/π_t'	prices associated to balances of pallets with products on distribution center t before and after the arrival of a replenishment convoy
π_p^e/π_t^e	prices associated to balances of pallets with recovered goods on plant p/distribution center t

Binary variables

X_r	variable denoting that route r belongs to the optimal subset of feasible routes.
S_{ij}	variable for sequencing the locations i and j along a route
Y_i	variable used to determine that the site i belongs to the route designed by a slave routes-generator problem

Continuous variables

CV	cost of the pick-up and delivery route designed by the slave routes-generator problem
C_i	traveling cost to reach the customer site i
L_i	quantity of pallets with recoverable goods collected up to customer i
T_i	time spent to reach the customer site i
TV	time spent on the route designed by the slave routes-generator problem
U_i	quantity of pallets full of products delivered up to customer i

goods back to plants while LTL trucks usually visit customers to fulfill their demands and to collect reusable resources. In the process of designing a distribution network, the response time, the product availability and the customer satisfaction are usually the main issues to take into account. To effectively design and manage a distribution network, the long-run strategic planning, the medium-term tactical planning and the short-term operational planning are periodically updated. Operational planning is concerned with short-term production scheduling, inventory management, and transportation planning (Dondo et al., 2009). Transportation represents a significant component of the total logistics cost and the short-term transportation planning generates vehicle routes based on the available resources in order to minimize the transportation cost while meeting some service-level requirements. This paper deals with the operational planning of flows of products and recoverable goods along a typical multi-echelon distribution and recovering network and introduces a decomposition framework devised to allow the efficient computation of these flows.

The paper is organized as follows: in Section 2 the literature related to the researched subject is reviewed and in Section 3 the most common distribution networks are briefly described. In Section 4 the integrated forward and reverse logistic problematic is described, formalized and mathematically formulated. A decomposition algorithm based in the branch-and-price paradigm to solve the problem is detailed in Section 5. Computational results are presented in Section 6 and the Section 7 presents the conclusions of the work.

2. Literature review

The development of computational tools for logistics management has attracted a great attention from both industry and academia to the field of supply chain management and to the field of EWO. Since these tools bring some methodologies developed in the operations research (OR) field to the process-system-engineering (PSE) area, we first review some PSE papers and later some OR contributions that are useful in the PSE and EWO fields.

Strategic and tactical planning of supply chain networks have been a very active research area over the past two decades. Reviews about the subject are Vidal and Goetschalckx (1997), Varma et al.

(2007), and Papageorgiou (2008). Operational planning of multi-echelon distribution networks, started to receive attention later. Among others, the following supply chain management problems related to the chemical industry were reported in the literature: distribution of refined petroleum products through a fleet of multi-parcel trucks (Van der Bruggen et al., 1995); scheduling of multi-compartment chemicals carriers (Jetlund and Karimi, 2004; Coccola et al., 2015); dispatching of industrial gases from plants to consumers (You et al., 2011; Marchetti et al., 2014); and collection of milk from farms and distribution of dairy products to outlets (Tarantilis and Kiranoudis, 2007). Research about distribution on multi-echelon logistic networks has focused on integrated logistics models for locating production plants and distribution facilities. In addition to strategic decisions, these approaches simultaneously determine the product mix to be manufactured and the product flow from factories to customer zones via intermediate warehouses. See for example Sarmiento and Nagir (1999) and Tsiakis et al. (2001). The last authors proposed a mixed-integer linear programming (MILP) formulation for the design of multi-product, multi-echelon supply networks but the number, location, and capacity of warehouses, and the flows of products throughout the system were pre-fixed. So, decisions were constrained just to the distribution of products. Jayaraman and Ross (2003) developed a two-level planning approach for distribution problems involving a central manufacturing site, multiple hubs and cross-docking sites, and customer zones. At the upper-level, the best set of depots and cross-docks is selected and at the lower level, the product flows from the plant to distribution centers, and the quantity transhipped to cross-dock sites and distributed to retail outlets are determined. Both steps are fulfilled by sequentially solving two MILP models. You and Grossmann (2008) developed an integrated approach that simultaneously finds the supply chain network, the planning schedule, and the inventory management under demand uncertainty. In order to identify the best network-design, a Pareto-optimal curve is generated by solving a multi-period MINLP formulation that simultaneously maximizes the net present value and minimizes the expected response time. Amaro and Barbosa-Povoa (2008) presented an integrated MILP formulation for the optimal scheduling of supply chain networks. The model provides an operational plan by considering the supply chain topology, different operational conditions, and market opportunities. Zeballos et al. (2014) proposed a multi-layer approach for addressing multi-period, multi-product closed-loop supply chains with uncertain levels of raw material supplies and of customer demands. The consideration of a multi-period setting leaded to a multi-stage stochastic programming problem handled by an MILP formulation. The effects of uncertainty on demands and on supply on the network were considered by means of multiple scenarios. Niknejad and Petrovic (2014) studied the optimization of integrated reverse logistic networks with different routes and developed a fuzzy mixed integer programming model which facilitates decision making in presence of uncertainty in demand and quantity of returned products of different quality levels. Mallidis et al. (2014) presented a methodology for assessing emissions in a multi-echelon logistics network which serves a geographical region through a number of retail stores. Several alternative network realizations were assessed. Kalaitzidou et al. (2015) introduced a mathematical programming framework that employs a generalized supply-chain-network coupled with forward and reverse logistics activities. The work solves by branch-and-bound a multi-product, multi-echelon and multi-period MILP problem in a closed-loop supply-chain network.

In the OR field, the N-echelon location routing problem (NE-LRP), the inventory routing problem (IRP) and the vehicle routing problem (VRP) are well-known tactical-level distribution problems. Most NE-LRP studies are related to two-echelon systems. The aim of the NE-LRP is to define the structure of the network by optimizing the number and location of facilities in both echelons, the fleet size for each level and the material flow distribution on each echelon. On the other hand, the IRP is a planning problem that provides a starting point for integrating inventory management and transportation. Its objective is to minimize the average distribution costs over the planning horizon, while avoiding stock outs. The VRP is the problem of satisfying, from a central depot the demands of a given number of customers with a given fleet of capacitated vehicles at minimum cost and several variations of this problem exist. These variants are currently solved by an effective decomposition technique called column generation. Since there is a large body of OR literature on these problems, we just refer to the very detailed survey written by Drexl and Schneider (2015) about LRP, the review by Cuda et al. (2015) about multi-echelon routing problems and the tutorial paper by Lübbecke and Desrosiers (2005) as well as the book by Desaulniers et al. (2005) about column generation and its applications in vehicle routing and in another fields. The two-echelon vehicle routing problem (2E-VRPCD) was introduced by Perboli et al. (2011) as an extension of the classical VRP. In the 2E-VRPCD freight delivery from a single depot to customers is managed by routing and consolidating the load at intermediate depots called satellites. Afterwards, the freight is sent from satellites to customers. The problem assumes a main depot, and a fixed number of capacitated satellites. Customer demands are fixed and known in advance and must be satisfied within the scheduling horizon. The objective is the minimization of the total transportation cost. In the general case of the 2E-VRPCD, each satellite can be served by more of than one vehicle and, therefore, the satellite demand can be split. In the second level, each customer must be served by just a single vehicle. The authors proposed a MILP formulation with valid cuts to get strong lower bounds to solve the problem. Dondo et al. (2009) introduced the so-called vehicle routing problem in supply chain management (VRP-SCM) which is a generalization of the N-echelon vehicle routing problem. The allocation of customers to suppliers and the quantities of products shipped from each source to a client are model decisions. However, the approach cannot handle cross-docking operations. So, Dondo et al. (2011) introduced a MILP formulation for the N-echelon multi-item vehicle routing problem with cross-docking and time windows (NE-VRPCD) that can be regarded as a generalization of the VRP-SCM. In the NE-VRPCD several products are handled and customer demands involving more than one item can be satisfied through either direct shipping or via intermediate facilities. Transshipment operations are triggered when a product-stock in a warehouse is insufficient to meet the demand of the assigned customers and the inventory at the end of the planning horizon. Supplies may come from factories or from other warehouses and the distribution strategy is found by solving the MILP formulation through a branch-and-cut algorithm.

3. Configuration of multi-echelon networks for forward and reverse logistic activities

As inbound flows to distribution centers are usually transported on full large-trucks, while outbound flows to customers are transported on smaller delivery-trucks, the number of distribution centers has a large effect on a network's transport efficiency as an increase on this number can significantly reduce outbound transportation distances (Dekker et al., 2012). The delivery of freight is often managed through one or more intermediate facilities where storing and consolidation activities are performed. This type of distribution systems is called multi-echelon and each echelon refers to one level of the distribution network. In addition, freight transportation can be broadly categorized into two classes according to the presence of one or more intermediate facilities; *direct shipping*

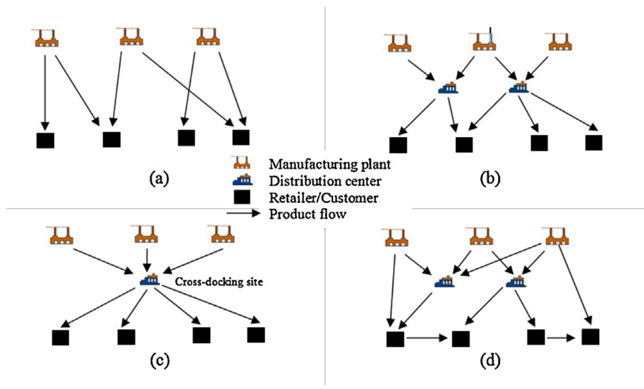

Fig. 1. Distribution network configurations (Dondo et al., 2009).

takes place when freight is directly delivered and *indirect shipping* takes place when freight, or part of the freight, is moved through intermediate facilities before reaching destination. As the network configuration strongly affects the supply chain performance by setting the framework within which operational decisions are taken, several network topologies arise. The usual configuration of distribution networks are (a) *manufacturer storage and direct shipping*, (b) *distributor storage and shipping* via *distribution centers*, (c) *cross-docking with manufacturer storage and shipments via a central depot*, and (d) *hybrid networks* (Dondo et al., 2009). These configurations are depicted in Fig. 1.

The integration of forward and reverse logistic activities leads to several possibilities. Because of the reverse flow of recycled goods back from customers must be taken into account, the utilization of the same transportation facilities for forward and backwards flows is considered as the most efficient option. Hybrid networks combining *manufacturer storage* and *distributor storage* are the most favored ones mostly because they are able to receive and consolidate both the forward and reverse flows in the same facilities. The concept of "hybrid network" refers here to the combined used of manufacturer storage and distributor storage for the forward and reverse flows whose quantities must be optimally fixed. In the remaining of this paper, a hybrid network utilized to fulfill products demands via direct and indirect shipping and to recover recyclable goods from customers is studied and modeled. The network quite accurately represents the downside side of a company supply-chain distributing standardized containers full of products and recycling standardized containers loaded with re-utilizable goods as e.g. reusable bottles. The network sketched in Fig. 2 illustrates the ways products and recoverable goods are moved between the echelons of the network.

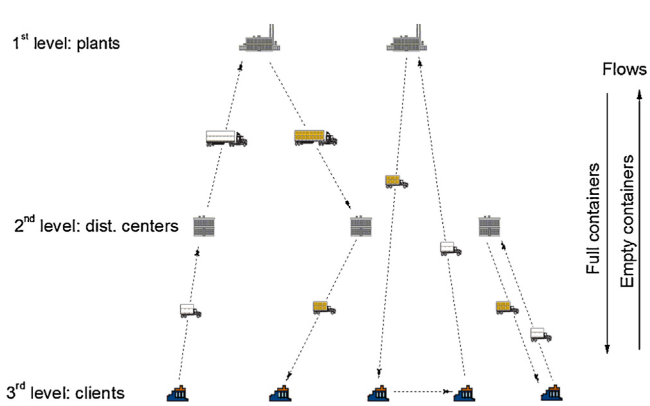

Fig. 2. A hybrid 3-echelon logistic network used on for integrated forward and backwards distribution and recovering activities.

The assumptions considered to mathematically model it are the following:

1. Data about demands, stocks, time windows and costs are deterministic and known in advance.
2. Several products can be palletized and transported by a vehicle but the containers are considered as a single commodity. The words 'container' and 'pallet' must be loosely understood as a standardized transport unit.
3. There are no predefined suppliers for customers and demands can be met by products coming from alternative sources. Also there is not a priori pre-fixed destination for recyclable goods recovered from customers.
4. The loading of pallets with recovered goods and unloading of units full of products are carried out on the customer location. The vehicle stop time-length has two components, a fixed stop-time and a variable component proportional to the amount of pallets to pick-up and/or to deliver.
5. Pick-up and delivery activities must be fulfilled before a maximum activities time.
6. Time windows may be assigned to the start time of pickup and delivery services.
7. Vehicles costs include two terms: a fixed vehicle-utilization cost and a variable cost related to the traveled distance.
8. A separate class of large vehicles, featuring its own capacity, its own fixed and variable transportation costs, is used to move cargo from plants to distribution centers and to return recyclable goods from there to plants.
9. The flow pattern of pallets with products through the distribution network (i.e., the number of vehicles moving goods from plants to distribution centers, the number and length of vehicle tours devoted to the fulfillment of delivery activities and the allocation of customers to servicing vehicles) is defined by the problem solution. The same assumption applies to the pick-up of recyclable goods and to the reverse flow of pallets with recovered goods.
10. The objective is to minimize the sum of vehicles fixed costs, traveling costs and transshipping costs for fulfilling all distribution and recollection activities.

The above described problem is formally defined in the next section.

4. Formulation of the problem

Let us consider a logistic network represented by a graph G $(P \cup T \cup I, A)$. Nodes $p \in P$ stand for production plants, $t \in T$ stand for distribution centers, and $i \in I$ stand for retailers and customers. A is the network of minimum-cost arcs interconnecting plants, distribution centers and customers in the network. The plants $p \in P$ and the distribution centers $t \in T$ can be regarded as source nodes or suppliers because they provide products to demanding locations $i \in I$. The elements of T and I are destination nodes for product shipments; I is a set of source sites of transport units with recyclable goods and T can behave as destiny and source for these units. The network allows the following transportation options: (i) direct shipping from plants to customers, (ii) direct shipping from distribution centers to customers, and (iii) indirect shipping from plants via distribution centers. All options may be used to channelize the forward (backward) flows from (to) the plants. Associated to every arc $a \in A$, there are a distance-based traveling cost c_{ij} and a travel time t_{ij}. Vehicles moving products from a plant/distribution center to consumers have a stated transport capacity (q) and a fixed utilization cost (cf_v). This plant/distribution center behaves as base from which the vehicles start and end the designed tour.

Large trucks used for moving products from plants to distribution centers and for transporting recyclable goods from distribution centers to plants have a capacity q_0 and a fixed utilization cost c_0. A replenishment order usually includes one or more than one FTL shipments from a stated plant to a given distribution center. Conversely, a recovering order defines the shipment of one or more than one FTL shipment from a given distribution center to a stated plant. Visits to customers must start within a stated time windows $[t_i^{\min}, t_i^{\max}]$ for all customers $i \in I$. Service times st_i at each customer location $i \in I$ have two components: a fixed preparation time t_f and a variable component that depends on the size of the load to pick-up/deliver and so $st_i = t_f + t_v\ |l_i|$ where t_v is the time spent in loading/unloading a transportation unit. The number of trips, the plants and distribution centers from where trips start/end and the forward and backwards dispatches between plants and distribution centers must be determined by the problem solution. The operational costs depend on the number of pick-up and delivery routes, on the number of replenishment/recovery dispatches, and on the number of incurred cross-docking operations. The objective is to minimize the sum of cross-docking costs, vehicles fixed costs and vehicles traveling costs while satisfying the following operational constraints: (i) all customers must be visited just once and only by one vehicle. (ii) The service at each customer must start within its time window. (iii) Each route begins at a plant/distribution center and ends at the same plant/distribution center. (iv) The sum of quantities of pallets with products and with recyclable goods onboard the vehicle must not exceed its transport capacity. (v) Replenishment dispatches start at the beginning of the planning horizon $t = 0$ and end on distribution centers at time $t = t_0$. The time-length t_0 include the times spent on loading operations on plants, on the trip and on the unloading operations on distribution centers. (vi) Pickup and delivery routes may start at any time with the planning horizon but must be fulfilled within the time interval [0, $t^{\max}$]. (vii) Recovering dispatches going from distribution centers to plants start after the closure of the servicing time horizon $t^{\max}$.

In order to formulate the above formalized problem as an Integer Program, let us denote R as the set of pickup and delivery routes, R^+ as the set of replenishment routes and R^- as the set of reverse transportation routes. For each route $r \in R$, c_r denote its cost, given by the sum of the costs of the arcs traveled by the vehicle plus the fixed vehicle-utilization-cost. Routes $r \in \{R^+ \cup R^-\}$ represent direct trips of large trucks between plants and distribution centers and include the cost of the associated cross-docking operations at plants/distribution centers in addition to fixed and traveling truck costs. Let us assume we are given a binary parameters a_{ir} indicating whether route $r \in R$ visits ($a_{ir} = 1$) or not ($a_{ir} = 0$) the location $i \in I$. For a route $r \in R$, let us consider also a binary parameter b_{rp}/b_{rt} that takes value 1 if route r starts/end on the plant p/distribution center t and 0 otherwise and nonnegative parameters α_r/β_r denoting the quantity of transport units to move from(to) the plants/distribution centers. We use the binary decision variable X_r to determine if the route $r \in \{R^T \cup R^+ \cup R^-\}$ belong to the optimal solution or not. So, the problem can now be formulated as:

Minimize

$$\sum_{r \in R^+} c_r X_r + \sum_{r \in R^-} c_r X_r + \sum_{r \in R} c_r X_r \tag{1}$$

Subject to:

$$\sum_{r \in R} a_{ir} X_r = 1 \quad \forall i \in I \tag{2}$$

$$\sum_{r \in R} b_{pr} \alpha_r X_r + \sum_{r \in R^+ :\, b_p^{start} = 1} q_0 X_r \le i_p^0 \quad \forall p \in P \tag{3}$$

$$\sum_{r \in R:\, t_r^{start} < t^0} b_{tr} \alpha_r X_r \le i_t^0 \quad \forall t \in T \tag{4}$$

$$\sum_{r \in R} b_{tr} \alpha_r X_r \le i_t^0 + \sum_{r \in R^+ :\, b_t^{end} = 1} q_0 X_r \quad \forall t \in T \tag{5}$$

$$\sum_{r \in R} b_{pr} \beta_r X_r + \sum_{r \in R^- :\, b_p^{end} = 1} q_0 X_r \ge \delta_p^f \quad \forall p \in P \tag{6}$$

$$\delta_t^0 + \sum_{r \in R} b_{tr} \beta_r X_r \ge \sum_{r \in R^- :\, b_t^{start} = 1} q_0 X_r \quad \forall t \in T \qquad X_r \in \{0, 1\} \tag{7}$$

The objective function (1) minimizes the total cost, i.e., the cost of all kind of routes. Constraint (2) assures that customer $i \in I$ is visited exactly once while the products availability constraint on each plant $p \in P$ is enforced by constraint (3). Eq. (4) is a products availability constraint for each distribution center $t \in T$ assuring that the quantity of cargo delivered from this terminal by vehicles departing before the arrival of the re-supply convoys cannot be higher than the initial inventory on the distribution center. Eq. (5) states a balance constraint of pallets full of products on each distribution center $t \in T$. It enforces that the total quantity of cargo delivered from there cannot be higher than the sum of the initial inventory plus the cargo unloaded on the distribution center. Constraints (6) and (7) are similar to constrains (4) and (5) but enforce balances of pallets with recovered goods on plants and on distribution centers respectively. So, by Eq. (6), the sum of quantities of pallets directly delivered to a plant plus the quantity of units delivered there via a stop on distribution centers must be at least as big as its demand of pallets with recyclable goods. By Eq. (7), the quantity of pallets with recovered-goods delivered from distribution center $t \in T$ cannot be higher than the sum of its pallets inventory plus the quantity of pallets unloaded on the distribution center.

5. The solution algorithm

In this section, the model defined by Eqs. (1)–(7) is embedded into a decomposition procedure in order to generate solutions for the problem above formulated. The formulation represents the set of all feasible routes $r \in \{R^+ \cup R^- \cup R\}$ and its objective is to select the minimum-cost subset of routes such that all customer demands are fulfilled and all plant requests of recovered goods are satisfied. Since the number of distribution centers and plants is much smaller than the number of customers $i \in I$ and because of the forward and backward trips involve the use of a single arc, replenishment/recovery trips can be totally enumerated. It is not possible to generate all feasible pick-up and delivery routes but a column generation approach can handle this complexity by implicitly considering all of them through the solution of a linear relaxation of the formulation (1)–(7). To do this, a subset of feasible pick-up and delivery routes (usually an initial but suboptimal solution) is enumerated and the linear relaxation of problem (1)–(7) is solved considering just this subset. The problem is called the reduced master problem (RMP) and the linear relaxation, the relaxed RMP. The solution is used to determine if there are routes $r \in R$ not included in the subset that can reduce the objective function value of the RMP. Using the values of the optimal dual variables for the constraints of the master problem with respect to this partial routes-set, new route are generated and incorporated into the columns pool, and the linear relaxation of the RMP is solved again. It is worth noting that the linear RMP relaxes just variables X_r for all routes $r \in R$ but variables related to replenishment and recovery trips $r \in \{R^+ \cup R^-\}$ remain integer. That is because a relaxation of variables related to these enumerated trips would lead to solutions with fractional replenishment and recovery trips strictly satisfying, as equality, constraints (3)–(7) but without a meaning in the real world. In the end, due to the fact that replenishment and recovery trips have already been enumerated and cannot be generated, the relaxed RMP is computed in two stages; an integer program is first solved, branching just on variables X_r for all $r \in \{R^+ \cup R^-\}$ to fix the optimal replenishment and recovery trips and later this information is feed to the relaxed RMP to compute the duals for constraints (2)–(7).

The dynamic activation of replenishment and recovering trips lead to a more complex procedure that is out of the scope of this work. So, we considered more practical just to include the small set of all replenishment and recovery trips in the initial RMP. The procedure iterates between the master problem and a slave routes-generator problem until no pick-up and delivery routes with negative reduced costs can be found. Afterwards, an integer master problem may be solved for finding the best subset of routes. Since the solution found may not be the global optimal, it is necessary to start a branching procedure to find it. So, the column generation procedure must be embedded into a branch-and-bound tree because some routes not generated when solving the relaxed RMP of the root node may be needed to find the best subset of routes. In summary, the procedure starts with an feasible solution and decomposes the problem into a master–slave structure comprising a relaxed RMP with the optimal replenishment and recovery trips that relaxes variables X_r for all routes $r \in R$ and the slave tour-generator problem. The master–slave structure is recursively solved until no feasible routes $r \in R$ can be generated. In that case, the relaxed RMP is solved again to verify the solution integrality. If the solution to this problem is integer, the optimal solution to the distribution and recovering problem has been found and the procedure ends. Otherwise; the integer solution to the RMP or global upper bound (GUB) will have a value higher than the value of the solution to the relaxed RMP or global lower bound (GLB). Therefore, the procedure starts branching according to a branching rule in order to generate the missing routes. At each tree-node, the mechanism is repeated and the bounds are compared. If the local lower bound (LLB), given by the value of the relaxed RMP, is higher than the GUB, (given by the available best integer solution) the node is fathomed; otherwise it is divided into two child-nodes that are included in the database of unsolved subspaces. Afterwards, the next subspace is fetched from the database to repeat the mechanism until this base is empty or until all nodes have been fathomed. Finally, the solution is specified by solving, for each selected column, a traveling salesman problem with time windows. The whole process is named *branch-and-price* and involves the definition of the relaxed RMP, the definition of the *slave routes-generator* or *pricing problem* and the implementation of a *branching rule*.

5.1. The master problem

The integer formulation (1)–(7) contains a number of binary variables which grows with the number of routes $r \in R$ of the pool. In order to compute a lower bound to its objective function value, we first solve an integer program to set the value of variables X_r for all $r \in \{R^+ \cup R^-\}$ while relaxing the integrality condition for variables X_r for all $r \in R$ and consider a feasible solution comprising the selected replenishment and recovery routes plus a small set of routes departing from plants and distribution centers to visit customers. Although a strictly exact procedure should compute columns for every feasible combination of replenishment and recovery trips, this two stages procedure is computationally cheaper and works very well in practice. With frozen values of variables X_r for all $r \in \{R^+ \cup R^-\}$, the constraints (2)–(7) are reordered in order to give rise to the following relaxed RMP:

Minimize

$$\sum_{r \in R^+} c_r X_r + \sum_{r \in R^-} c_r X_r + \sum_{r \in R} c_r X_r \tag{1}$$

Subject to:

$$\sum_{r \in R} a_{ir} X_r = 1 \quad \forall i \in I \tag{2.b}$$

$$\sum_{r \in R} b_{pr} \alpha_r X_r \le i_p^0 - \sum_{r \in R^+ : b_p^{start}=1} q_0 X_r \quad \forall p \in P \tag{3.b}$$

$$\sum_{r \in R:\, t_r^{start} < t^{transf}} b_{pr} \alpha_r X_r \le i_t^0 \quad \forall t \in T \tag{4.b}$$

$$\sum_{r \in R} b_{pr} \alpha_r X_r \le i_t^0 + \sum_{r \in R^+ : b_t^{end}=1} q_0 X_r \quad \forall t \in T \tag{5.b}$$

$$\sum_{r \in R} b_{pr} \beta_r X_r \ge \delta_p^f - \sum_{r \in R^- : b_p^{end}=1} q_0 X_r \quad \forall p \in P \tag{6.b}$$

$$\sum_{r \in R^- : b_t^{start}=1} q_0 X_r \le d_t^0 + \sum_{r \in R} b_{pr} \beta_r X_r \quad \forall t \in T \qquad 0 \le X_r (r \in R) \le 1 \tag{7.b}$$

The left-hand side of Eqs. (2.b)–(7.b) contains terms related to the relaxed variables X_r for all $r \in R$ and the right hand side contains the constant terms related to the frozen configuration of replenishment and recovery trips. After finding the optimal solution to this relaxed RMP, the dual variables values for constraints (2.b)–(7.b) are available and can be passed to the pricing problem in order to produce more profitable routes $r \in R$ to later reduce the value of the objective function (1). At each iteration, the used configuration of replenishment and recovery trips is re-evaluated, the linear relaxation of the RMP is first solved and afterwards new columns with a negative reduced cost are generated by solving the pricing problem to be next detailed.

5.2. The pricing problem

Each pick-up and delivery tour is an elementary path from a plant/distribution center to the same plant/distribution center through some customers on the network. The pricing problem is an elementary shortest path problem with resource constraints (ESPPRC) and is NP-hard in the strong sense. The most used technique to solve pricing problems was dynamic programming in which a relaxation of the pricing algorithm was solved and cycles were allowed. Nevertheless, the recent trend relies in algorithms in which pricing problems are exactly solved without allowing cycles. In our application we solve the MILP formulation of the elementary pricing problem with a branch-and-cut solver trying to generate several solutions per iteration.

The objective of the slave problem is to find a route that minimize the quantity stated by Eq. (8) and is subject to the constraints stated by Eqs. (9)–(20):

Minimize

$$CV - \sum_{i \in I^+} \pi_i Y_i - i_p \sum_{p \in P} x_p \pi_p - i_t (1 - i_{t'}) \sum_{t \in T} x_t \pi_t - i_t \sum_{t \in T} x_t \pi'_t - i_p \sum_{p \in P} x_p \pi_p^e - i_t \sum_{t \in T} x_t \pi_t^e \tag{8}$$

Subject to

$$C_i \geq \sum_{p \in P} x_p c_{pi} + \sum_{t \in T} x_t c_{ti} \quad \forall i \in I \tag{9}$$

$$\left\{ \begin{array}{l} C_j \geq C_i + c_{ij} - M_C(1 - S_{ij}) - M_C(2 - Y_i - Y_j) \quad (10.a) \\ C_i \geq C_j + c_{ij} - M_C S_{ij} - M_C(2 - Y_i - Y_j) \quad (10.b) \end{array} \right\} \quad \forall i, j \in I : i < j$$

$$C \geq cf_v + C_i + \sum_{p \in P} x_p c_{pi} + \sum_{t \in T} x_t c_{ti} - M_C(1 - Y_i) \quad \forall i \in I \tag{11}$$

$$T_i \geq t_0 i_t i_{t'} + \sum_{p \in P} x_p t_{pi} + \sum_{t \in T} x_t t_{ti} \quad \forall i \in I \tag{12}$$

$$\left\{ \begin{array}{l} T_j \geq T_i + st_i + t_{ij} - M_T(1 - S_{ij}) - M_T(2 - Y_i - Y_j) \quad (13.a) \\ T_i \geq T_j + st_j + t_{ji} - M_T S_{ij} - M_T(2 - Y_i - Y_j) \quad (13.b) \end{array} \right\} \quad \forall i, j \in I : i < j$$

$$T \geq T_i + \sum_{p \in P} x_p t_{pi} + \sum_{t \in T} x_t t_{ti} - M_C(1 - Y_i) \quad \forall i \in I \tag{14}$$

$$t_i^{\min} \leq T_i \leq t_i^{\max} \quad \forall i \in I \tag{15}$$

$$T \leq t^{\max} \tag{16}$$

$$\left\{ \begin{array}{l} L_j \geq L_i + \gamma_j - M_L(1 - S_{ij}) - M_L(2 - Y_i - Y_j) \quad (17.a) \\ L_i \geq L_j + \gamma_i - M_L S_{ij} - M_L(2 - Y_i - Y_j) \quad (17.b) \\ U_j \geq U_i + \varepsilon_j - M_L(1 - S_{ij}) + M_L(2 - Y_i - Y_j) \quad (17.c) \\ U_i \geq U_j + \varepsilon_i - M_L S_{ij} + M_L(2 - Y_i - Y_j) \quad (17.d) \end{array} \right\} \quad \forall i, j \in I : i < j$$

$$\left\{ \begin{array}{l} L_i \leq \sum_{i \in I} \gamma_i Y_i \quad (18.a) \\ U_i \leq \sum_{i \in I} \varepsilon_i Y_i \quad (18.b) \end{array} \right\} \quad \forall i \in I$$

$$q - U_i + L_i \geq 0 \quad \forall i \in I \tag{19}$$

$$\left\{ \begin{array}{l} \sum_{i \in I} \gamma_i Y_i \leq q \quad (20.a) \\ \sum_{i \in I} \delta_i Y_i \leq q \quad (20.b) \end{array} \right\} \quad \forall i \in I$$

Since we use a single pricing problem to generate routes starting from plants and from distribution centers, and for tours that depart from distribution centers before and after the arrival of a replenishment convoy, we use some binary parameters to define the starting location and to freeze some decisions. We just refer to Dondo and Cerdá (2006) for an explanation about MILP formulations that made a joint use of binary variables and decision parameters on MILP routing formulations. The objective function of the slave problem is the cost of the generated route minus some collected dual-prices associated to constraints (2.b)–(7.b). Binary parameters $i_p = 1$ and $i_t = 1$ are used to state that the tour departs from a plant or from a distribution center respectively. The parameter $i_{t'}$ indicates that a vehicle departs before ($i_{t'} = 0$) or after ($i_{t'} = 1$) the arrival of a replenishment convoy to a distribution center. Duals π_i are associated to constraints (2.b) and duals π_p are associated to balances constraints on plants (3.b). The duals π_t and π'_t are associated to constraints (4.b) and (5.b) stating balances on distribution centers of transport units full of products before and after the arrival of a replenishment convoy. Duals associated to balances (6.b) and (7.b) of transport units with recovered goods on plant and on distribution centers are stated by $\pi_p{}^e$ and $\pi_t{}^e$ respectively. The binary parameters x_p/x_t indicate the departing and ending plant/distribution center of the designed tour. The constraint (9) set the minimum distance to reach the customer $i \in I$ as the distance of going directly from the plant p/distribution center t to the location i. Eqs. (10) fix the accumulated distance up to each visited site. If locations i and j are both allocated onto the generated route ($Y_i = Y_j = 1$), the visiting ordering for both sites is determined by the sequencing variable S_{ij}. If location i is visited before j ($S_{ij} = 1$), according constraints (10.a), the traveling cost up to the location j (C_j) must be larger than C_i by at least c_{ij}. In case node j is visited earlier, ($S_{ij} = 0$), the reverse statement holds and constraint (10.b) becomes active. If one or both sites are not allocated to the tour, Eq. (10) become redundant. Eq. (11) computes the route-cost CV by the addition of the fixed vehicle utilization cost cf_v to the traveling-cost up to the site to which the vehicle must return after visiting the last customer. Since the last visited customer-location cannot be known before the problem resolution, Eq. (11) must be written for each customer $i \in I$. M_C is an upper bound for the variable CV. Constraint (12) states the departure time of the vehicle from plant p/distribution center t just in case the tour starts there. The timing constraints stated by Eqs. (13)–(14) are similar to constraints (10)–(11) but they apply to the time dimension. M_T is an upper bound for the times T_i spent to reach the nodes $i \in I$ and for the tour-time-length TV. Eq. (15) forces the service time on any customer $i \in I$ to start at a time T_i bounded by the time window $[t_i^{\min}, t_i^{\max}]$. Eq. (16) sets the maximum routing time for vehicles returning to its allocated plant/distribution center. Load-based sequencing constrains are handled by Eq. (17). Let us assume that customers $(i, j) \in I$ are both serviced along the same tour ($Y_i + Y_j = 2$). If node i is visited before j ($S_{ij} = 1$), then the total quantity of pallets loaded on the vehicle up to node j (L_j) should be larger than L_i by at least ε_j, where the parameter ε_j is equal to the quantity of pallets with recoverable goods available on customer j. Such a condition is enforced by Eq. (17.a). If customer j is visited earlier, then Eq. (17.b) holds and L_i must be larger than L_j by at least ε_i. In addition, if node i is visited before j, then the quantity of pallets full of products unloaded up to node j (U_j) should be larger than U_i by at least γ_j (the parameter γ_j is equal to the quantity of pallets full of products demanded by customer $j \in I$); so, Eq. (20.c) activates. If node j is visited earlier ($S_{ij} = 0$), the reverse condition holds and Eq. (20.d) would become active. If one or both nodes are not on the tour, then $Y_i + Y_j < 2$ and all constraints (17) become redundant. M_L is a large positive number. Upper bounds for variables L_i and U_i are stated by Eqs. (18.a) and (18.b) respectively. Eq. (19) is a capacity constraint for loads onboard the vehicle. Eqs. (20.a) and (20.b) are capacity constraints for the total quantity of pallets with products and with recovered goods loaded/unloaded on the vehicle, respectively.

5.3. A branching strategy

The linear relaxation of the RMP may not be integer and applying a standard branch-and-bound procedure to this problem with a given pool of columns may not yield an optimal solution (Barnhart et al., 2000). A non-generated column pricing favorably may exist but it may not be present in the RMP. Consequently, if the relaxation of the RMP is not integer, to find the optimal solution, columns must be generated after branching. Ryan and Foster (Ryan and Foster, 1981) proposed a branching rule that fits in a natural way with the above formulated slave-problem. The rule amounts to selecting two customers i and j and generating two branch-and-bound nodes; one in which i and j are serviced by the same vehicle and the other where they are serviced by different vehicles. Many practical applications have constraints like Eq. (2) and Ryan and Foster observed, that for a feasible integer solution to a problem with this structure, each constraint contains one variable with value 1 in the solution and all other variables contained in that constraint must have value 0. When regarding two different constraints, they are either fulfilled by the same route, i.e., a route that is satisfying both constraints, or by different tours. In this case, one variable has solution value set to 1 that is contained in the first constraint but not in the second one and one variable has value set to 1 that is contained in the second, but not in the first one. We call the DIFFER-child the one that, by fixing all variables to zero, cover two customers i and j and they will be fulfilled by two different vehicles with solution value 1. If both customers have to be fulfilled by the same vehicle we call this child the SAME-child. To enforce this branching constraint, rather than adding it explicitly them to the master problem, it is easier just to delete columns from the columns-set considered in the branch-and-price node and to enforce them at the pricing level either by eliminating the customer(s) from the graph and/or by forcing the tour to visit the fixed customer(s).

5.4. Implementation and numerical issues

The incomplete branch-and-price algorithm has been coded in GAMS 23.6.2 and integrates a column generation routine comprising the computation of the relaxed RMP detailed on Section 5.1 and the slave-problem of Section 5.2 into a branch-and-bound procedure that branches according to the rule described in Section 5.3. Both GAMS routines were built over routines developed by Kalvelagen (2009, 2011) and integrated in this work to lead to the decomposition procedure. Minor branching and assembling modifications were introduced in both routines. They were aimed at replacing the NLP of the Kalvelagen (2009) MINLP algorithm by the column generation procedure detailed in Section 5.2 and at forbidding the infeasible branching combination $Y_i = 0$ for all $i \in I$. Since there are customers that cannot be in the same tour because of time-windows incompatibilities and/or because of long distances between them, the branching rule must branch o couples of customers that are 'compatible'; i.e., that can be visited by a single vehicle. The algorithm uses the CPLEX 11 as the MILP sub-algorithm for generating columns and for computing upper and lower bounds. As the relaxed RMP is computed after the setting of the best configuration of replenishment and recovery trips, we must previously solve an IP branching on variables X_r for all routes $r \in \{R^+ \cup R^-\}$ to find the optimal replenishment/recovery configuration. Since the number of enumerated routes $r \in \{R^+ \cup R^-\}$ is small this integer program can be easily solved in a fraction of a second. Since branch-and-price is an enumeration algorithm enhanced by fathoming based on bound comparisons, to work with the strongest bounds should be the best option, although the mechanism can work with any bound. The best upper bound may need the resolution of an integer RMP while the best node lower-bound can be obtained by solving the relaxed RMP just after the column generation sub-algorithm was unable to produce one more profitable column. So, the best bounds may mean a higher computational cost than weaker bounds. This leads to a trade-off between the CPU time used in computing strong bounds and the size of the exploration-tree. This, in turn, leads to the use of some standard strategies (Desaulniers et al., 2002) to improve the overall computational performance. In this way, to reduce the "tailing-off" effect consisting in a very low convergence-rate at the last iterations of the master–slave recursion we ended it after 10 iterations in no-root nodes and used the bounds computed in such a way. Time-windows reduction and pre-processing were also used in order to increase the resolution efficiency. Although branching according the Ryan and Foster rule tend to generate unbalanced trees, we choose this rule because of simplicity of implementation. Since a small number of variables of the master problem will have a positive value in an optimal solution, most of the variables will be 0. When fixing a variable to 0 in the integer master problem this has in most cases no big effect, since this variable is likely to have value 0 in the optimal solution, anyway. Fixing it to 1 has more effect since one of the small set of variables that have a strictly positive value is determined. This may lead to an unbalanced branch-and-bound tree. More efficient but more complex and cumbersome branching rules can be implemented but this is out of the scope of this work. Every time the column generation procedure ends, the node lower bound is computed and its integrality checked. If the solution is integer and better than the GUB, then it replaces this bound. If not, fast integer solutions are searched and provided by the GUROBI solver (just in case a tree-node LLB is not integer). Since we collect several routes per master–slave iteration via the `SolnPool` procedure (CPLEX Solver Manual, 2007), we implemented a solutions filter aimed at deleting non-optimal solutions that visit the same customers but in different order. So, after the resolution of the pricing problem, we feed the RMP with just the best tour visiting a given subset of customers. Due to the use of the two-stages method to compute the relaxed RMP, and because the maximum number of nodes to inspect is bounded and the master–slave recursion is terminated after 10 iterations in no-root nodes, this procedure is a heuristic that focus in efficiency rather than in proving optimality. The node selection strategy is 'best first search', that means selecting the node with the best LLB of the pool of unsolved subspaces. The algorithm run in a 2-cores 2.5-Ghz 6-Mbytes RAM Notebook and the mechanism settings used to solve the problems are summarized in Table 1.

Table 1
Setting options of the algorithm.

Option	
MILP solver (slave problems)	CPLEX 11
MILP solver (optional GUB computing problems)	Gurobi
Branching rule	Ryan and Foster
Nodes selection strategy	Best first search
Maximum CPU time per master–slave iteration (s)	30
Multiple columns generated per iteration	Yes
Filtering of columns visiting the same subset of customers	Yes
Time-windows reduction	Yes
Maximum number of master-slave iterations per b&p node	10 (no-root nodes)
Maximum number of branch-and-price inspected nodes	100
Master problem	Partitioning
Columns pool	Up to 10,000

To provide an initial solution, feasible routes p–i–p starting from any plant $p \in P$ and feasible routes t–i–t starting from any distribution center $t \in T$ are generated and associated to each customer $i \in I$. From this initial routes-package plus the set of all forward/backward routes $r \in \{R^+ \cup R^-\}$, the configuration of replenishment/recovery trips is fixed and the linear RMP can be computed to start the master–slave recursion.

6. Results and discussion

The procedure was first used to solve a few small examples to illustrate how different stock-levels and demands of recoverable goods change the topology of the best routes. Later, it was applied to several instances generated from a case study with the aim of testing the capability of the procedure to bring good solutions to realistic problems.

6.1. An illustrative small example

To illustrate how the inventories and demands levels change the optimal trip configurations, let us consider a very small example involving 2 plants (P_1, P_2), 2 distribution centers (T_1, T_2) and 5 customers ($i_1, \ldots, i_5$) whose locations in the Euclidean plane are given by the Fig. 3. The sites locations were chosen to make the solutions intuitive. Customers' demands of standardized pallets with products and customer's stocks of standardized pallets with recoverable goods are also stated in the figure. Euclidean distances between the locations are computed from location coordinates and a fixed vehicle utilization cost $cf = 20$ Euclidean units is added to the to the cost of any vehicle tour. Traveling times are considered numerically equal to the traveled distance and each customer service time is computed as *st = 10 time units + 0.5 (demand + stock) time units*.

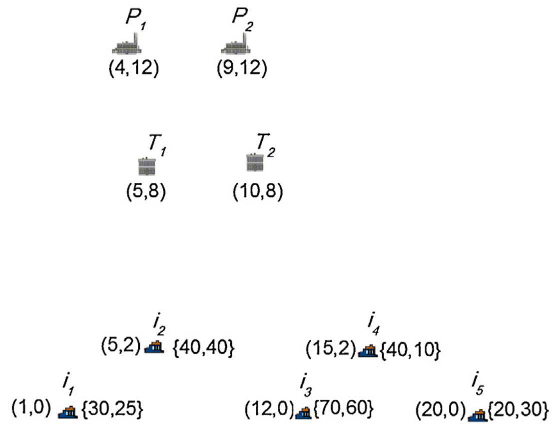

Fig. 3. Plants, terminals and customer's locations for the illustrative example (coordinates in parenthesis; demands and stocks in brackets).

Let us consider a first instance with a products-stock of 200 pallets on plants P_1 and P_2 and a products-stock of 100 pallets on distribution centers T_1 and T_2. No demands of recoverable goods from plants are still considered in this case. The optimal solution was found in 1.6 s and implies a total cost of 151 Euclidean units. Now let us consider the scenario on which the stock on terminal T_1 is null and it remains in 100 units for T_2. The solution to this case was recomputed in 1.2 s and implied a cost of 158 Euclidean units. In the next instance, let us consider a demand of 100 pallets with recovered goods for plant P_1. This demand raises the cost of the optimal solution up to 164 Euclidean units. In the last illustrative instance, we consider and additional demand of 150 pallets for plant P_2 and stocks of 50 recovered pallets both in T_1 and T_2. This demand forced the use of two transfer trips between terminals and plants and raises the cost of the optimal solution up to 239 Euclidean units. The solution was found in 3.3 s. All solutions are summarized in Table 2 and depicted in Fig. 4. It is clear that minor changes in demands and stocks can strongly change the topology of solutions.

Table 2
Solution to the instances of the illustrative example.

Tour	α	β	*CV*	*TV*
Instance 1				
P2-n5-n4-P2	40	40	62.0	73.5
T1-n2-n1-T1	70	65	40.3	76.3
T2-n3-T2	70	60	48.7	89.5
Total cost (IS)				151
LS				146
CPU time (s)				1.6
Instance 2				
P1-n2-n1-P1	70	65	53.4	902
T2-n5-n4-T2	60	40	54.7	735
P2-n3-P2	70	60	49.8	763
Total cost (IS)				158
LS				154
CPU time (s)				1.2
Instance 3				
P1-n2-n1-P1	70	65	55.9	902
T2-n5-n4-T2	60	40	57.4	735
P1-n3-P1	70	60	53.6	901
Total cost (IS)				164
LS:				164
CPU time (s)				3.3
Instance 4				
P2-n3-P2	70	60	54.3	86.7
P2-n5-n4-P2	60	40	57.4	90.7
P1-n2-n1-P1	70	65	55.9	101.9
T1-P1			35.9	
T2-P2			35.9	
Total cost (IS)				239
LS				239
CPU time (s)				3.3

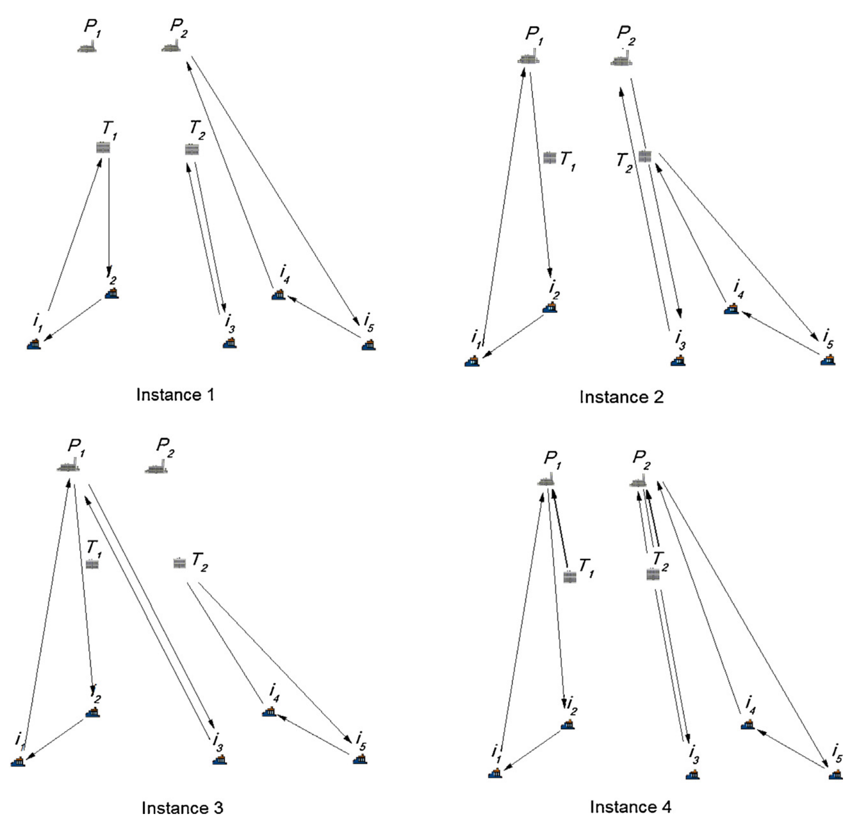

Fig. 4. Solution to the instances of the illustrative example.

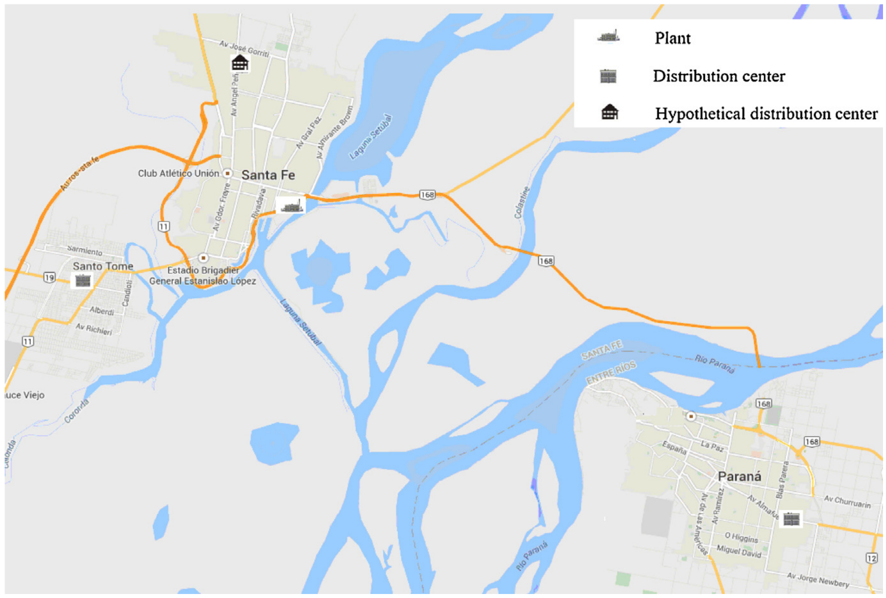

Fig. 5. Geographical view of the distribution and recollection area.

Table 3
A sensitivity analysis on fixed costs and on the activation of replenishment and recovery trips.

Fixed cost	S. Fe stock	S. Tomé stock	Paraná stock	S. Fe demand	S. Tomé demand	Paraná demand	Integer solution ($)	Linear solution ($)	Σ_r *cf* ($)	Σ_r *CV* ($)	Tours[a]	CPU (s)
	5000	50	100	0	0	0						
0							888	870	0	888	7-0-0	43.7
50							1238	1168	350	888	7-0-0	41.4
100							1588	1447	700	888	7-0-0	57.1
200							2288	1997	1400	888	7-0-0	68.7
	5000	50	100	200	300	300						
0							898	875	0	898	8-0-0	42.9
50							1304	1182	350	954	7-0-0	41.7
100							1654	1465	700	954	7-0-0	56.1
200							2354	2023	1400	954	7-0-0	80.2
	5000	50	70	200	300	300						
0							1404	1375	0	1404	8-1-0	41.5
50							1881	1756	425	1456	7-1-0	61.2
100							2311	2114	850	1465	7-1-0	86.0
200							3165	2822	1700	1465	7-1-0	99.8
	5000	50	70	400	300	300						
0							1561	1532	0	1561	7-1-1	61.1
50							2060	1978	500	1560	7-1-1	62.4
100							2560	2407	1000	1560	7-1-1	93.4
200							3560	3257	2000	1560	7-1-4	105.6

[a] Pick-up and delivery tours-replenishment trips-recovery trips.

6.2. *Testing instances*

We test now the solution procedure on several instances of different sizes generated from a case study with real data. A beer production company has its main plant located in Santa Fe (Argentina) and from there several products are distributed to customers on the Santa Fe (SF) urban area and also to some customers in two nearby cities, Santo Tome (ST) and Paraná (P). Fig. 5 depicts the geographical area where distribution and recollection activities are carried out. The daily distribution and recovering operation involves the use of urban trucks starting tours from the main plant and from both distribution centers. These trucks have a maximum capacity $q = 300$ pallets and are used to deliver pallets with full bottles and to recover pallets with empty bottles. By 'pallet' we mean a standard box-container able to carry up to twelve bottles. The vehicle stop-times on customer sites have two components; a fixed-contribution (15′), and a variable component that is proportional to the amount of pallets to pick-up and to deliver (0.5′/pallet).

Time windows usually are not considered but sometimes they can be assigned just for a few customers. Distance (in km) between clients and between these locations and plants/distribution centers are estimated by using the Manhattan distance formula jointly with sites locations on the cities map. The average urban-travel speed is conservatively assumed to be 20 km/h and from the distance between locations and from such a speed, the traveling times are computed. Larger FTL trucks are used to replenish distribution centers with pallets loaded with full bottles and to collect back pallets with re-utilizable bottles. These trucks have a transport capacity $q = 1200$ pallets. Although a fraction of this load may be necessary to fulfill the demand allocated to a distribution center, the load that is not subsequently delivered remains inventoried on the distribution center for the next distribution campaign. The time to load pallets, travel from the plant to the distribution centers and unload pallets

Table 4a
Solutions overview for instances without time-windows.

Customers			Forward flow			Backward flow			Solution parameters							
|SF|	|ST|	|P|	Santa Fe stock	Santo Tomé stock	Paraná stock	Santa Fe demand	Santo Tomé stock	Paraná stock	Integer solution ($)	Linear solution ($)	Gap[a] (%)	CPU (s)	Cols.	Pickup delivery tours	Replenish. trips	Recovery trips
13	3	5	5000	50	100	0	0	0	1262	1168	7.46	41.4	174	7	0	0
13	3	5	5000	50	100	200	300	300	1310	1182	9.82	41.7	175	7	0	0
13	3	5	5000	50	70	200	300	300	1881	1756	6.67	61.2	209	7	1	0
13	3	5	5000	50	70	400	300	300	2060	1978	3.94	62.4	211	7	1	1
18	4	6	5000	200	100	0	300	300	1526	1482	2.85	177.0	317	8	0	0
18	4	6	5000	200	100	300	300	300	1612	1527	5.27	176.2	318	8	0	0
18	4	6	5000	100	50	300	300	300	2192	2110	3.77	211.9	333	8	1	0
18	4	6	5000	15	25	400	300	300	2594	2548	1.76	191.7	328	8	2	1
30	5	10	5000	100	100	0	0	0	2714	2621	3.41	2060.2	596	11	1	0
30	5	10	5000	100	100	500	300	300	2860	2697	5.71	2571.1	687	12	1	0
30	5	10	5000	40	50	1000	300	300	3748	3607	3.76	3472.5	659	15	2	1
52	12	18	5000	5000	5000	0	0	0	3496	3317	5.14	5363.1	1197	19	0	0
52	12	18	5000	100	200	0	0	0	4302	4131	4.27	10,308.4	1380	18	2	0
52	12	18	5000	100	100	500	300	300	4365	4176	4.34	10,090.6	1400	18	2	0
52	12	18	5000	40	50	1000	300	300	4616	4475	3.05	9339.8	1294	19	2	1

Table 4b
Solutions overview for instances with time-windows.

Customers			Forward flow			Backward flow			Solution parameters							
|SF|	|ST|	|P|	Santa Fe stock	Santo Tomé stock	Paraná stock	Santa Fe demand	Santo Tomé stock	Paraná stock	Integer solution ($)	Linear solution ($)	Gap[a] (%)	CPU (s)	Cols.	Pickup delivery tours	Replenish. trips	Recovery trips
13	3	5	5000	50	100	0	0	0	1276	1195	6.36	39.8	189	7	0	0
13	3	5	5000	50	100	200	300	300	1355	1208	10.8	40.9	190	7	0	0
13	3	5	5000	50	70	200	300	300	1921	1782	7.22	69.2	220	7	1	0
13	3	5	5000	50	70	400	300	300	2099	2005	4.48	71.7	221	7	1	1
18	4	6	5000	200	100	0	300	300	1555	1501	2.95	133.5	318	8	0	0
18	4	6	5000	200	100	300	300	300	1643	1554	5.42	134.4	318	8	0	0
18	4	6	5000	100	50	300	300	300	2223	2138	3.85	179.3	333	8	1	0
18	4	6	5000	15	25	400	300	300	2626	2576	1.87	208.0	341	8	2	1
30	5	10	5000	100	100	0	0	0	2783	2680	3.71	1595.5	579	11	1	0
30	5	10	5000	100	100	500	300	300	2916	2753	5.61	1950.0	615	12	1	0
30	5	10	5000	40	50	1000	300	300	3797	3664	3.49	2916.8	706	14	2	1
52	12	18	5000	5000	5000	0	0	0	3571	3377	4.43	5833.9	1287	19	0	0
52	12	18	5000	100	200	0	0	0	4375	4250	2.86	7559.3	1347	18	2	0
52	12	18	5000	100	100	500	300	300	4364	4218	3.35	6041.5	1169	18	2	0
52	12	18	5000	40	50	1000	300	300	4587	4521	1.43	6505.1	1211	19	2	1

[a] Gap = 100 (integer solution–linear solution)/integer solution.

is assumed as $t_0 = 90'$. The datasheet used to generate instances is presented in the Appendix and utilizes operational information from a typical working day. Instances are generated by selecting the first $|n|$ clients of each town. For example, |SF| means we took the first Santa Fe clients from the list of the Appendix. Stocks of pallets with full bottles and demands of pallets with empty bottles for plants and distribution centers are also problem parameters. The vehicles departing from the plant and from the distribution centers can visit customers located at most 12 km from there. The 'zonification' practice used to delimit overlapped services zones, which is useful to reduce the solution space, from a practical point of view, is used to forbid the possibility of customers e.g. from Santo Tomé to be serviced by a vehicle warehoused on Paraná.

The first aim was to perform a brief sensitivity analysis about the influence of fixed costs and of activation of replenishment and recovery trips on total costs. So, we generated several instances for a small example with |SF| = 13, |ST| = 3 and |P| = 5 by varying both the vehicles utilization costs and the level of stocks and demands. The results are presented in Table 3. From the information of this table it seems that bigger fixed costs enlarge the gap between the integer solution and the linear solution. The gap remains in spite that the algorithm is unable to produce more profitable columns. The activation of replenishment and recovery trips does not seems to change the integrality gap but these trips tend to complicate the problem resolution as it can be observed in the rise of the CPU time consumed to compute solutions to instances that use such trips. It seems that a fixed cost more or less similar to half the incurred traveling cost is enough to minimize the number of used vehicles. Afterwards, a series of instances involving from 21 to 82 customers with different stocks and demands have been solved. A fixed truck utilization cost cf_v = \$50 and a unit distance cost \$10/km are considered to compute the pick-up and delivery tour-costs. For larger vehicles traveling replenishment and recovery trips, the fixed vehicle utilization cost is cf_0 = \$ = 75 and the unit-distance traveling cost is \$15/km.

Table 4a summarizes the found results. This table lists the number of clients to service in each city, the availability of pallets with full bottles and pallets with recovered bottles on plants and on distribution centers and the demand of the pallets with recovered bottles by the plant. The table also reports information related to the found solutions. Later the same instances we re-solved but with the time windows reported in the Appendix and the numerical results are reported in Table 4b. The following observations can be derived from the information of both tables: (i) for a given example-size, the CPU time consumed to compute a solution grows with the size of the replenishment and recovery flow. (ii) The integrality gap is quite independent of the problem size and of the stock sizes. The biggest gap is 7.46% in instances without time-windows and 7.22% in instances with time windows. (iii) For a given example-size, tight product stocks on distribution centers increase the number of pick-up and delivery tours. This is caused by the necessity to wait for the arrival of replenishment convoys in order to satisfy demands allocated to the distribution center. So, some vehicles deliver cargo already inventoried but some vehicles must wait to receive cargo transshipped from replenishment tour to later visit their allocated customers. (iv) As expected, time windows constraints lead to smaller gaps and smaller CPU times because these constraints are useful in reducing the solution space. The CPU time was reduced roughly by a third with respect to instances without time windows.

Now, let us consider the hypothetical opening of a new depot in the north of Santa Fe in order to satisfy the nearby demand. Furthermore, let us consider the Paraná distribution center as a plant because of an un-constraining product-stocks there and because we want to observe how the tours patterns and the forward and reverse flow all change if this location is utilized for disposing

Table 5
Solutions overview for instances comprising two plants (Santa Fe and Paraná) and two distribution centers (Santo Tomé, Santa Fe 2).

Customers			Forward flow				Backward flow				Solution parameters							
|SF|	|ST|	|P|	Santa Fe stock	Santo Tomé stock	Paraná stock	Santa Fe 2 stock	Santa Fe demand	Paraná demand	Santo Tomé stock	Santa Fe 2 stock	Integer solution (\$)	Linear solution (\$)	Gap[a] (%)	CPU (s)	Cols	Pickup delivery tours	Replenish. trips	Recovery trips
52	12	18	5000	5000	5000	5000	0	0	0	0	3020	2896	4.16	6382.1	1480	19	0	0
52	12	18	5000	100	5000	100	0	0	0	0	3446	3350	2.79	10,593.5	1773	18	2	0
52	12	18	5000	150	5000	120	450	400	300	300	3906	3744	4.15	12,020.8	1597	21	2	0
52	12	18	5000	80	5000	70	500	600	600	600	4359	4196	3.74	17,527.4	1836	20	2	1
52	12	18	5000	50	5000	80	750	750	600	600	4462	4196	6.03	23,274.0	1850	20	2	2
Instances with time windows																		
52	12	18	5000	5000	5000	5000	0	0	0	0	3169	2996	5.45	6325.7	1530	18	0	0
52	12	18	5000	100	5000	100	0	0	0	0	3589	3450	3.87	10,426.0	1728	18	2	0
52	12	18	5000	5000	5000	5000	450	400	300	300	3997	3826	4.28	11,021.9	1724	21	2	0
52	12	18	5000	5000	5000	5000	500	600	600	600	4409	4282	2.88	15,208.8	1770	20	2	1
52	12	18	5000	5000	5000	5000	750	750	600	600	4585	4388	4.49	17,812.8	1896	20	2	2

[a] Gap = 100 (integer solution–linear solution)/integer solution.

Table 6
Solution to the large instance with 750 units of demand of recovered pallets from both plants and with time windows.

		Delivered pallets	Recovered pallets	Start time (′)	End time (′)	Cost ($)
Replenishment trips						
Santa Fe–Santo Tomé		1200		0	90	258
Santa Fe–Santa Fe 2		1200		0	90	205
Base: Santa Fe plant						
1	Santa Fe – sf46-sf43-sf11 – Santa Fe	220	108	0	86	59
2	Santa Fe – 44-sf15-sf16-sf47-sf42 – Santa Fe	252	236	0	183	227
3	Santa Fe – sf22-sf23-sf41-sf8 – Santa Fe	296	228	0	179	186
4	Santa Fe – sf51-sf35-sf14-sf13-sf25-sf45 – Santa Fe	292	280	0	302	185
5	Santa Fe – sf49-sf29-sf26-sf12-sf30 – Santa Fe	260	280	0	191	148
6	Santa Fe – sf31-sf10-sf24-sf3-sf42 – Santa Fe	296	272	0	305	188
7	Santa Fe – sf20-sf32-sf40-sf39-sf37-sf48 – Santa Fe	228	200	0	231	179
8	Santa Fe – sf50-sf28-sf38 – Santa Fe	200	180	0	140	107
Base: Santa Fe 2 distribution center						
1	Santa Fe 2-sf52-sf1-sf21-sf9-sf7 – Santa Fe 2	276	228	0	167	163
2	Santa Fe 2-sf4-sf36-sf19 – Santa Fe 2	284	184	90	210	126
3	Santa Fe 2-sf34-sf17-sf18-sf2 – Santa Fe 2	300	180	90	271	142
4	Santa Fe 2-sf6-sf5-sf27-sf33 – Santa Fe 2	268	228	90	268	138
Base: Santo Tomé distribution center						
1	Santo Tomé – s3-s4-s10 – Santo Tomé	280	200	90	251	123
2	Santo Tomé – s8-s9-s12-s7 – Santo Tomé	284	272	90	346	157
3	Santo Tomé – s2-s11-s1-s5-s6 – Santo Tomé	252	212	90	251	168
Base: Paraná distribution and recycling center						
1	Paraná – p10-p9-p18-p7-p8 – Paraná	300	220	0	311	222
2	Paraná – p11-p15-p6-p14-p1-p4 – Paraná	296	280	0	319	277
3	Paraná – p3-p16 – Paraná	128	72	0	202	196
4	Paraná – p17-p5 – Paraná	200	200	0	161	141
5	Paraná – p12-p2-p13 – Paraná	120	140	0	136	142
Recovery trips						
	Santo Tomé–Paraná	1200			360	643
	Santa Fe 2–Santa Fe	1200			360	205
Total costs						$4585

and recycling re-utilizable bottles. This reconfiguration changes the network topology that now includes the set of plants $P = \{\text{Santa Fe, Paraná}\}$ and the distribution center set $T = \{\text{Santo Tomé, Santa Fe2}\}$. We solved some of the largest instances for this new network configuration with and without time windows and summarized the results in Table 5.

The observation of the information summarized in Table 5 lead to conclusions similar to the above derived from Table 4. Logically, the more complex network lead to an increase of the CPU time consumed to compute solutions. In the end, it is clear that tight inventories on distribution centers lead to an increase of routing costs and should be avoided. To illustrate the level of information provided by the procedure, we detail in Table 6 the solution to the instance with $|SF| = 52$, $|ST| = 12$, $|P| = 18$, a demand of 750 recovered units on both plants and time-windows. This solution involves 8 trips from the Santa Fe plant, 4 trips from the hypothetical north distribution center, 3 trips from the Santo Tomé distribution center and 5 trips from the Paraná distribution and recycling center.

Note that 6 trips start after $t = 90'$ because, due to tight inventories, the trucks must wait to load pallets previously transported by replenishment trucks. Also, 2 replenishment trucks depart from the plant and go repetitively to the Santo Tomé and to the Santa Fe 2 distribution centers. After the end of pick-up and delivery trips, a large truck move pallets with recovered bottles from Santo Tomé to Paraná and another large truck moves pallets from the north distribution center to the main Santa Fe plant. The solution was found in 17800 s and implies a $4585 total cost while the integrality gap is 4.49%.

7. Conclusions

A decomposition procedure based on the paradigm of column generation has been developed for computing the forward and reverse flow of standardized containers with products and with recycled goods on a three-echelon logistic network. The procedure aims at designing the best set of distribution and recovering routes and to select the cheapest replenishment and recovering trips connecting plants and distribution centers in an integrated logistic network. In such a network some products can be delivered via direct shipping, from a plant or from a regional depot, or from a plant via an intermediate stop in a distribution center. The recovery of standardized containers with re-utilizable goods also can be fulfilled via direct and indirect shipping. The integrated logistic network accounts for a given transportation infrastructure to design routes interconnecting factories and distribution centers in order to replenish them and to bring back recyclable goods to plants. The unloading of large quantities of cargo from FTL trucks on distribution centers to later distribute them by smaller trucks allows to lowering transportation costs through an efficient use of truck capacities. Moreover, the studied problem considers the reverse flow of containers with recyclable goods by using the same trucks and facilities. The sources to satisfy customer's demands rather than being fixed data are free variables to be fixed by the problem solution. In this way the concept of "transportation request" usual in the transportation research area is here replaced by sets of cargo-source and cargo-sink locations. This concept allows considering a more realistic view of nowadays logistic networks of chemical and food industries. The problem was first modeled as a set partitioning problem with an additional set of transferring and balances constraints and later the model was embedded into a decomposition procedure based in branch-and-price concepts. The proposed mechanism has been used to solve numerous instances featuring different levels of products-stocks on plants and on distribution centers. The model is quite complex but, with the given algorithm-settings, the solutions were obtained in reasonable CPU times and show integrality gaps bellow 10% in all but one instance.

Some issues may be improved in a future research. For example, from an OR perspective, an explanation about the integrality gap that remains (even if the algorithm is unable to produce more profitable routes) should be found. This gap seems to depend on the weight of fixed costs on vehicle tours. From a EWO perspective, new strategies to fulfill services to customers and to perform replenishment and recovery tasks should be proposed and researched. For example, the option of multiple visits to customers (split pick-up and/or delivery) in order to lowering the operational costs should be explored. Another topic to consider is a time-varying inventory of products on production plants due to the incoming of recently fabricated products. This leads to research about the integration of inventory management, products delivery and recyclables recovery in the downside side of the supply chain.

Acknowledgments

Financial support from Universidad Nacional del Litoral (U.N.L.) under grant CAI+D 501 201101 00315 LI and from CONICET is greatly appreciated.

Appendix.

Supply sites locations.

Santa Fe plant	Santo Tomé depot	Paraná depot	Santa Fe 2 depot
$X_{coord} = 31$; $Y_{coord} = 17$;	$X_{coord} = -25$; $Y_{coord} = -6$;	$X_{coord} = 162$; $Y_{coord} = -41$;	$X_{coord} = 18$; $Y_{coord} = 59$;

Clients datasheet

Clients	Location		Demand	Stock to return	Time windows	
	X_{coord}	Y_{coord}			t_i^{min}	t_i^{max}
Santa Fe						
sf1	38	38	48	40	0	360
sf2	55	55	72	40	240	270
sf3	44	44	40	60	240	270
sf4	55	55	120	80	0	360
sf5	63	63	60	80	0	360
sf6	54	54	80	20	0	360
sf7	50	50	88	20	0	360
sf8	24	24	40	40	150	210
sf9	36	36	80	80	0	360
sf10	41	41	96	72	0	360
sf11	17	17	100	60	0	360
sf12	28	28	80	80	120	210
sf13	32	32	40	60	0	360
sf14	34	34	20	40	0	360
sf15	37	37	80	40	0	360
sf16	37	37	72	80	0	360
sf17	51	51	80	20	0	360
sf18	54	54	48	60	0	360
sf19	58	58	80	80	120	210
sf20	16	16	24	32	0	360
sf21	35	35	40	48	0	360
sf22	41	41	96	72	0	360
sf23	42	42	80	96	0	360
sf24	43	43	40	20	120	210
sf25	30	30	72	40	210	240
sf26	30	30	72	60	0	240
sf27	65	65	80	80	0	360
sf28	20	20	40	48	0	360
sf29	25	25	48	40	0	360
sf30	23	23	40	60	150	210
sf31	37	37	80	100	0	360
sf32	4	4	40	20	0	360
sf33	60	60	48	48	240	270
sf34	46	46	100	60	0	360
sf35	39	39	80	40	0	360
sf36	57	57	84	24	0	360
sf37	15	15	40	40	0	360
sf38	16	16	60	72	0	360
sf39	14	14	32	20	0	360
sf40	10	10	72	48	120	210
sf41	49	49	80	20	0	360
sf42	35	35	40	20	0	360
sf43	17	17	80	40	0	360
sf44	20	20	40	72	0	360
sf45	17	17	40	60	240	300
sf46	17	17	40	8	0	360
sf47	34	34	20	24	0	360
sf48	17	17	20	40	180	240
sf49	18	18	20	40	0	360
sf50	22	22	100	60	0	360
sf51	38	38	40	40	120	210
sf52	42	42	20	40	0	360

Santa Fe plant	Santo Tomé depot		Paraná depot		Santa Fe 2 depot	
$X_{coord}=31$; $Y_{coord}=17$;	$X_{coord}=-25$; $Y_{coord}=-6$;		$X_{coord}=162$; $Y_{coord}=-41$;		$X_{coord}=18$; $Y_{coord}=59$;	
			Clients datasheet			
Clients	Location		Demand	Stock to return	Time windows	
	X_{coord}	Y_{coord}			t_i^{min}	t_i^{max}
			Santo Tomé			
s1	−19	10	20	32	0	360
s2	−27	7	40	32	0	360
s3	−34	−8	80	80	0	360
s4	−34	−1	88	60	0	360
s5	−18	6	40	48	120	210
s6	−20	3	112	20	0	360
s7	−26	−15	96	100	0	360
s8	−32	−12	60	40	150	210
s9	−35	−17	80	72	0	360
s10	−30	−1	112	60	210	240
s11	−33	8	40	80	0	360
s12	−29	−20	48	60	270	330
			Paraná			
p1	138	−49	40	64	0	360
p2	170	−81	60	40	0	360
p3	145	−60	80	40	0	360
p4	168	−67	20	16	240	300
p5	150	−80	120	120	0	360
p6	142	−41	60	100	0	360
p7	155	−47	80	40	0	360
p8	159	−58	80	60	270	300
p9	168	−46	40	20	0	360
p10	170	−49	40	20	0	360
p11	152	−52	48	20	0	360
p12	169	−75	20	40	0	360
p13	157	−72	40	60	0	360
p14	136	−42	80	40	210	270
p15	150	−47	48	40	0	360
p16	140	−80	48	32	120	210
p17	163	−85	80	80	0	360
p18	161	−42	60	80	150	210

References

Amaro A, Barbosa-Povoa A. Optimal supply chain management with detailed scheduling. Ind Eng Chem Res 2008;47(1):116–32.

Barnhart C, Johnson E, Nemhauser G, Savelsbergh M, Vance P. Branch and price. Column generation for solving huge integer programs. Oper Res 2000;48(3):316–29.

CPLEX 12 Solver Manual. GAMS. The solver manuals; 2007.

Coccola M, Dondo R, Mendez C. A MILP-based column generation strategy for managing large-scale maritime distribution problems. Comput Chem Eng 2015;72:350–62.

Cuda R, Guastaroba G, Speranza MG. A survey on two-echelon routing problems. Comput Oper Res 2015;55:185–99.

Dekker R, Bloemhof J, Mallidis I. Operations research for green logistics – an overview of aspects issues and contributions. Eur J Oper Res 2012;219:671–9.

Desaulniers G, Desrosiers J, Solomon M. Accelerating strategies in column generation methods for vehicle routing and crew scheduling problems. Essays and surveys in metaheuristics. Oper Res/Comput Sci Interfaces Ser 2002;15:309–24.

Desaulniers G, Desrosiers J, Solomon M, editors. Column generation. New York: Springer; 2005.

Dondo R, Cerdá J. A MILP framework for dynamic vehicle routing problems whit time windows. Lat Am Appl Res 2006;36:255–61.

Dondo R, Mendez C, Cerda J. Managing distribution in supply-chain networks. Ind Eng Chem Res 2009;48(22):9961–78.

Dondo R, Méndez C, Cerdá J. The multi-echelon vehicle routing problem with cross-docking in supply chain management. Comput Chem Eng 2011;35(12):3002–24.

Drexl M, Schneider M. A survey of variants and extensions of the location-routing problem. Eur J Oper Res 2015;241:283–308.

Grossmann I. Advances in mathematical programming models for enterprise-wide optimization. Comput Chem Eng 2012;47:2–18.

Ilgin M, Gupta S. Environmentally conscious manufacturing and product recovery (ecmpro): a review of the state of the art. J Environ Manage 2010;91(3):563–91.

Jayaraman V, Ross A. A simulated annealing methodology to distribution network design and management. Eur J Oper Res 2003;144(3):629–45.

Jetlund AS, Karimi IA. Improving the logistics of multicompartment chemical tankers. Comput Chem Eng 2004;28(8):1267–83.

Kalaitzidou A, Longinidis P, Georgiadis M. Optimal design of closed-loop supply chain networks with multifunctional nodes. Comput Chem Eng 2015;80:73–91.

Kalvelagen E. Columns generation with GAMS; 2011, Available from: http://amsterdamoptimization.com/pdf/colgen.pdf.

Kalvelagen E. Some MINLP solution algorithms; 2009, Available from http://www.amsterdamoptimization.com/pdf/minlp.pdf.

Lübbecke M, Desrosiers J. Selected topics in column generation. Oper Res 2005;53(6):1007–23.

Mallidis I, Vlachos D, Iakovou E, Dekker R. Design and planning for green global supply chains under periodic review replenishment policies. Transp Res E 2014;72:210–35.

Marchetti PA, Gupta V, Grossmann I, Cook L, Valton P, Singh T. Simultaneous production and distribution of industrial gas supply-chains. Comput Chem Eng 2014;69:39–58.

Niknejad A, Petrovic D. Optimization of integrated reverse logistics networks with different product recovery routes. Eur J Oper Res 2014;238:143–54.

Papageorgiou LG. Supply chain optimization for the process industries: Advances and opportunities. In: Proceeding FOCAPO-2008; 2008. p. 33–8.

Perboli G, Tadei R, Vigo D. The two-echelon capacitated vehicle routing problem: models and math-based heuristics. Transp Sci 2011;45(3):364–80.

Ryan D, Foster B. An integer programming approach to scheduling. In: Wren A, editor. Computer scheduling of public transport urban passenger vehicle and crew scheduling. North-Holland, Amsterdam; 1981. p. 269–80.

Sarmiento AM, Nagir R. A review of integrated analysis of production–distribution systems. IIE Trans 1999;31(11):1061–74.

Tarantilis CD, Kiranoudis CT. A Flexible adaptive memory based algorithm for real-life transportation operations. Eur J Oper Res 2007;179(3):806–22.

Tsiakis P, Shah N, Pantelides CC. Design of multi-echelon supply chain networks under demand uncertainty. Ind Eng Chem Res 2001;40:3585–604.

Van der Bruggen L, Gruson R, Salomon M. Reconsidering the distribution structure of gasoline product for a large oil company. Eur J Oper Res 1995;81:460–73.

Varma VA, Reklaitis GV, Blau GE, Pekny JF. Enterprise wide modeling and optimizations. An overview of emerging research challenges and opportunities. Comput Chem Eng 2007;31:692–711.

Vidal CJ, Goetschalckx M. Strategic production-distribution models: a critical review with emphasis on global supply chain models. Eur J Oper Res 1997;98:1–18.

You F, Pinto JM, Capón E, Grossmann IE, Arora N, Megan L. Optimal distribution-inventory planning of industrial gases. I. Fast computational strategies for large-scale problems. Ind Eng Chem Res 2011;50:2910–27.

You F, Grossmann IE. Design of responsive supply chains under demand uncertainty. Comput Chem Eng 2008;32:3090–111.

Zeballos LJ, Méndez CA, Barbosa-Povoa AP, Novais AQ. Multi-period design and planning of closed-loop supply chains with uncertain supply and demand. Comput Chem Eng 2014;66:151–64.