CIRRELT

Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport

Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation

Optimal Joint Replenishment and Delivery of Perishable Products

Leandro C. Coelho
Gilbert Laporte

March 2013

CIRRELT-2013-20

Bureaux de Montréal :
Université de Montréal
C.P. 6128, succ. Centre-ville
Montréal (Québec)
Canada H3C 3J7
Téléphone : 514 343-7575
Télécopie : 514 343-7121

Bureaux de Québec :
Université Laval
2325, de la Terrasse, bureau 2642
Québec (Québec)
Canada G1V 0A6
Téléphone : 418 656-2073
Télécopie : 418 656-2624

www.cirrelt.ca

Université Laval · UQÀM · HEC Montréal · École Polytechnique Montréal · Université de Montréal

Optimal Joint Replenishment and Delivery of Perishable Products

Leandro C. Coelho[1,2,*], **Gilbert Laporte**[1,3]

[1] Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT)

[2] Department of Operations and Decision Systems, Université Laval, 2325, de la Terrasse, Québec, Canada G1V 0A6

[3] Department of Management Sciences, HEC Montréal, 3000, Côte-Sainte-Catherine, Montréal, Canada H3T 2A7

Abstract. Achieving the right balance between replenishing, keeping items in inventory or allowing some of them to spoil is a difficult problem when dealing with perishable products. In this paper we analyze the optimal joint decisions of when, how and how much to replenish customers with products of varying ages. The value of the products tends to decrease with their age, whereas their holding cost increases. We discuss the main features of problems arising in the joint replenishment and delivery of perishable products and we model them under general assumptions. We then solve the problem of jointly replenishing and delivering perishable products by means of an exact branch-and-cut algorithm, and we test its performance on a set of randomly generated instances. Our algorithm is capable of proving optimality for instances with up to 50 customers, three periods, and a maximum age of two periods for the perishable product, or up to 30 customers, six periods and a maximum age of three periods for the perishable product. For the unsolved instances the optimality gap is always less than 2.5% on average.

Keywords. Perishable products, inventory control, replenishment, optimization.

Acknowledgements. This work was partly supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant 39682-10. This support is gratefully acknowledged. We also thank Calcul Québec for providing high performance parallel computing facilities.

---

* Corresponding author: Leandro.Coelho@cirrelt.ca

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2013

1 Introduction

Inventory control constitutes an important logistics operation, especially when products have a limited shelf life. Keeping the right inventory levels guarantees that the demand is satisfied, without incurring unnecessary holding or spoilage costs. Several inventory control models are available [3], many of which include a specific treatment of perishable products [21].

Efficient delivery can provide further savings in logistics operations. The optimization of vehicle routes is one of the most developed fields in operations research [18]. The integration of inventory control and vehicle routing yields a complex optimization problem called inventory-routing, in which the aim is to minimize the overall costs related to vehicle routes and inventory control. Recent overviews of the inventory-routing problem (IRP) are those of Andersson et al. [2] and of Coelho et al. [8].

Problems related to the management of perishable products inventories arise in several areas. Applications of inventory control of perishable products include blood management and distribution [5, 9, 15, 16, 17, 24], radioactive and chemical products [1, 11, 27], and food such as dairy products, vegetables and fruits [4, 12, 22, 25, 26]. A review of the main models and algorithms in this area can be found in Nahmias [21].

The management of joint inventory and distribution of perishable products, which is the topic of this paper, gives rise to the perishable inventory-routing problem (PIRP). Hemmelmayr et al. [14] studied the case of blood inventory control with predetermined fixed routes and stochastic blood demand. The problem was solved heuristically by integer programming and variable neighborhood search. Custódio and Oliveira [10] proposed an strategical heuristic analysis of the distribution and inventory control of several frozen groceries with stochastic demand. Mercer and Tao [20] studied the weekly food distribution problem of a supermarket chain, without considering product age. A theoretical paper developing a column generation approach was presented by Le et al. [19] to provide solutions to a PIRP. The optimality gap was typically below 10% for instances with eight customers and five periods under the assumptions of fixed shelf life and flat value throughout the life of the product.

This paper makes several scientific contributions. We first classify and discuss the main assumptions underlying the management of perishable products. We then formulate the PIRP as a mixed integer linear program (MILP) for the most general case, and we present an exact branch-and-cut algorithm for its solution. To the best of our knowledge, this is the first time an IRP is modeled and solved exactly under general assumptions in the context of perishable products management. In particular, our model does not require any assumption on the shape of the product value and inventory cost functions. We also establish the relationships between the PIRP and the multi-product IRP recently studied by the authors [7].

The remainder of the paper is organized as follows. In Section 2 we describe the main assumptions of the problem under consideration and their treatment in our model. In Section 3 we present our MILP model, including new valid inequalities, followed by the proposed algorithm in Section 4. Computational experiments are presented in Section 5. In Section 6 we conclude our paper.

2 Problem Description

The joint replenishment and inventory problem for perishable products is concerned with the combined optimization of delivery routes and inventory control for products having a non-increasing value over time. These products typically have an expiry date, after which they are no longer good for consumption. This is the case of some law-regulated products such as food and drugs, but also of a wide variety of unregulated products whose quality, appearance or commercial appeal diminishes over time, such as flowers, cosmetics, paint, electronic products or fashion items. In this section we discuss four main assumptions underlying the treatment of these kinds of products, and we explain how we incorporate them in our model. Specifically, we discuss the types of product perishability in Section

2.1, the assumptions governing the inventory holding costs of these prodcuts in Section 2.2, their value as a functions of age in Section 2.3, and the management of items of different ages held in inventory in Section 2.4.

2.1 Types of product perishability

There exist two main types of perishable products according to how they decay [21]. The first includes products whose value do not change until a certain date, and then goes down to zero almost immediately. This is the case of products whose utility is no longer valued by the customers, such as year books, electronics or maps, which quickly become obsolescent after a given date or when a new generation of products enters the market. However, this is more a case of obsolescence than perishability. Even though these items may still be in perfect condition, they are simply no longer useful. Within the same category of products that maintain their appearance and usefulness, we find products with an expiry date, such as drugs, yogurt and bottled milk. These products can be consumed whether they are top fresh or a few days old, but after their expiry date, they are usually deemed unfit for sale by the retailers. The second type includes products whose quality or perceived value decays gradually over time due. Typical examples are fruits, vegetables and flowers.

2.2 The impact of item age on inventory holding costs

As a rule, the unit inventory holding cost changes with respect to the age and value of a product. This general assumption holds, for instance, in the case of insurance costs which are value related. In such cases, older items may yield a higher inventory holding cost than newer items. All the variable costs related to the age of the product can be modeled through a single parameter, called the unit inventory holding cost, which depends on the age of the item. In other cases, all items yield the same holding cost, regardless of their age. Products with short shelf life usually fit in this category. In this case, the holding cost, which encompasses all other variable costs, can be captured by a unique input parameter independent of the value and age of the product, which is the case in most applications.

2.3 Value of the item according to its age

A parameter that greatly affects the age of the item is its perceived value by consumers. Brand new items usually have a higher selling price, which decreases over time according to some function. In this paper we do not make any specific assumption regarding the shape of this function. Rather, we assume that the selling price is known in advance for each product age. Note that the function describing the relation between price and age can be non-linear, non-continuous or even non-convex, but it can still be accomodated by our model, as will be shown in Section 3.

2.4 Inventory management policies

The final assumption we discuss relates to the management of items of different ages held in inventory. From the previous assumption, we can safely assume that customers are indifferent between paying more for a newer item, or less for a used item. Hence, it is up for the retailer to decide which items to offer to customers, which will influence the associated revenue.

In such a context, three different policies can be envisaged. The first one consists of applying a last-in-first-out (LIFO) policy, i.e., the supplier always sells fresher products first. This policy ensures a longer shelf life and increases utility for the customers but, at the same time, yields a higher spoilage rate. The second policy is the reverse. Under a first-in-first-out (FIFO) rule, the items are sold in order of their arrival, which generates less spoilage, but also less revenue. The third policy, which we introduce in our model, is more flexible and lies between these two extremes. It lets the model determine which items to sell at any given time period in order to maximize profit. This means that depending on the parameter settings, one may prefer to spoil some items and sell fresher ones because they generate higher revenues, even if this means that holding costs will increase.

3 Mathematical Formulation

We now formally describe the mathematical formulation of PIRP under the assumptions just presented for a single product. The case of several products is conceptually similar, but requires an additional index [7]. We assume that the routing cost matrix is symmetric. Thus, we define the problem on an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{0, ..., n\}$ is the vertex set and $\mathcal{E} = \{(i, j) : i, j \in \mathcal{V}, i < j\}$ is the edge set. Vertex 0 represents the supplier and the remaining vertices $\mathcal{V}' = \mathcal{V} \setminus \{0\}$ represent n customers. A routing cost c_{ij} is associated with edge $(i, j) \in \mathcal{E}$.

Because of the general assumptions presented in Section 2, we consider that both the supplier and customers are fully aware of the number of items in inventory according to their age. This is important because the sales revenue and inventory holding costs are affected by the age of the product. The supplier has the choice to deliver fresh or aged product items, and each case yields different holding costs. Each customer has a maximum inventory holding capacity C_i, which cannot be exceeded in any period of the planning horizon of length p. At each time period $t \in \mathcal{T} = \{1, ..., p\}$, the supplier receives or produces a fresh quantity r^t of the perishable product. We assume the supplier has sufficient inventory to meet the demand of its customers during the planning horizon, and all demand has to be satisfied. At the beginning of the planning horizon the decision maker knows the current inventory level of the product at each age held by the supplier and by the customers, and receives information on the demand d_i^t of each customer i for each time period t. Note again that, as discussed in the previous section, the demand can be equally satisfied by fresh or aged products, which will in turn affect the revenue.

As is typically the case in the IRP literature [8], we assume that the quantity r^t made available at the supplier in period t can be used for deliveries to customers in the same period, and the delivery amount received by customer i in period t can be used to meet the demand in that period. A set $\mathcal{K} = \{1, \ldots, K\}$ of vehicles are available. We denote by Q_k the capacity of vehicle k. Each vehicle can perform at most one route per time period, visiting a subset of customers, starting and ending at the supplier's location. Also as in other IRP papers, we do not allow split deliveries, i.e., customers receive at most one vehicle visit per period.

The perishable product under consideration becomes spoiled after s periods, i.e., the age of the product belongs to a discrete set $\mathcal{S} = \{0, \ldots, s\}$. The product is valued according to its age, and the decision maker is aware of the selling revenue u_g of one unit of product of age g. Likewise, the inventory holding cost h_i^g in location $i \in \mathcal{V}$ is a function of the age g of the product. This general representation allows for flat or variable revenues, and for flat or variable holding costs depending on the age and value of the product, covering all situations described in Section 2.

The inventory level I_i^t held by customer i in period t comprises items of different ages. We break down this variable into $I_i^t = \sum\limits_{g \in \mathcal{S}} I_i^{gt}$, where I_i^{gt} represents the quantity of product of age h in inventory at customer i in period t. Likewise, we decompose the demand d_i^t into $\sum\limits_{g \in \mathcal{S}} d_i^{gt}$.

The aim of the problem is to construct vehicle routes for each period and to determine delivery quantities of products of different ages for each period and each customer, in order to maximize the total profit, equal to the sales revenue, minus the routing and inventory holding costs. This problem is extremely difficult to solve since it encompasses several NP-hard problems such as the vehicle routing problem [18] and a number of variants of the classical IRP [8].

Our MILP model works with routing variables x_{ij}^{kt} equal to the number of times edge (i, j) is used on the route of vehicle k in period t. We also use binary variables y_i^{kt} equal to one if and only if node i is visited by vehicle k in period t. Formally, variables $I_i^t = \sum\limits_{g \in \mathcal{S}} I_i^{gt}$

denote the inventory level at vertex $i \in \mathcal{V}$ at the end of period $t \in \mathcal{T}$, and d_i^{gt} denotes the quantity of product of age g used to satisfy the demand of customer i in period t, and we denote by q_i^{gkt} the quantity of product of age g delivered by vehicle k to customer i in period t. The problem can then be formulated as follows:

$$\text{(PIRP) maximize} \sum_{g \in \mathcal{S}} \sum_{t \in \mathcal{T}} u_i^g d_i^{gt} - \sum_{i \in \mathcal{V}} \sum_{g \in \mathcal{S}} \sum_{t \in \mathcal{T}} h_i^g I_i^{gt} - \sum_{(i,j) \in \mathcal{E}} \sum_{k \in \mathcal{K}} \sum_{t \in \mathcal{T}} c_{ij} x_{ij}^{kt}, \tag{1}$$

subject to

$$I_0^{gt} = I_0^{g-1,t-1} - \sum_{i \in \mathcal{V}'} \sum_{k \in \mathcal{K}} q_i^{gkt} \quad g \in \mathcal{S} \backslash \{0\} \quad t \in \mathcal{T} \tag{2}$$

$$I_0^{0t} = r^t \quad t \in \mathcal{T} \tag{3}$$

$$I_i^{gt} = I_i^{g-1,t-1} + \sum_{k \in \mathcal{K}} q_i^{gkt} - d_i^{gt} \quad i \in \mathcal{V}' \quad g \in \mathcal{S} \backslash \{0\} \quad t \in \mathcal{T} \tag{4}$$

$$I_i^{0t} = \sum_{k \in \mathcal{K}} q_i^{0kt} - d_i^{0t} \quad i \in \mathcal{V}' \quad t \in \mathcal{T} \tag{5}$$

$$\sum_{g \in \mathcal{S}} I_i^{gt} \leq C_i \quad i \in \mathcal{V}' \quad t \in \mathcal{T} \tag{6}$$

$$d_i^t = \sum_{g \in \mathcal{S}} d_i^{gt} \quad i \in \mathcal{V}' \quad t \in \mathcal{T} \tag{7}$$

$$\sum_{g \in \mathcal{S}} \sum_{k \in \mathcal{K}} q_i^{gkt} \leq C_i - \sum_{g \in \mathcal{S}} I_i^{g,t-1} \quad i \in \mathcal{V}' \quad t \in \mathcal{T} \tag{8}$$

$$q_i^{gkt} \leq C_i y_i^{kt} \quad i \in \mathcal{V}' \quad g \in \mathcal{S} \quad k \in \mathcal{K} \quad t \in \mathcal{T} \tag{9}$$

$$\sum_{i \in \mathcal{V}'} \sum_{g \in \mathcal{S}} q_i^{gkt} \leq Q_k y_0^{kt} \quad k \in \mathcal{K} \quad t \in \mathcal{T} \tag{10}$$

$$\sum_{j \in \mathcal{V}, i<j} x_{ij}^{kt} + \sum_{j \in \mathcal{V}, j<i} x_{ji}^{kt} = 2 y_i^{kt} \quad i \in \mathcal{V} \quad k \in \mathcal{K} \quad t \in \mathcal{T} \tag{11}$$

$$\sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}, i<j} x_{ij}^{kt} \leq \sum_{i \in \mathcal{S}} y_i^{kt} - y_m^{kt} \quad \mathcal{S} \subseteq \mathcal{V}' \quad k \in \mathcal{K} \quad t \in \mathcal{T} \quad m \in \mathcal{S} \tag{12}$$

$$\sum_{k \in \mathcal{K}} y_i^{kt} \leq 1 \quad i \in \mathcal{V}' \quad t \in \mathcal{T} \tag{13}$$

$$I_i^{gt}, d_i^{gt}, q_i^{gkt} \geq 0 \quad i \in \mathcal{V}' \quad g \in \mathcal{S} \quad k \in \mathcal{K} \quad t \in \mathcal{T} \tag{14}$$

$$x_{i0}^{kt} \in \{0,1,2\} \quad i \in \mathcal{V}' \quad k \in \mathcal{K} \quad t \in \mathcal{T} \tag{15}$$

$$x_{ij}^{kt} \in \{0,1\} \quad i,j \in \mathcal{V}' \quad k \in \mathcal{K} \quad t \in \mathcal{T} \tag{16}$$

$$y_{i}^{kt} \in \{0,1\} \quad i \in \mathcal{V} \quad k \in \mathcal{K} \quad t \in \mathcal{T}. \tag{17}$$

The objective function (1) maximizes the total sales revenue, minus inventory and routing costs. Constraints (2) define the inventory conservation conditions for the supplier, aging the product by one unit in each period. Constraints (3) ensure that supplier always produces or receives top fresh products. Constraints (4) and (5) define inventory conservation and aging of the items for the customers. Constraints (6) impose a maximal inventory capacity at each customers. Constraints (7) state that the demand of each customer in each period is the sum of product quantities of different ages. Note that by design, any product whose age g is higher than s is spoiled, e.g., it no longer appears in the inventory nor can it be used to satisfy the demand. Constraints (8) and (9) link the quantities delivered to the routing variables. In particular, they only allow a vehicle to deliver products to a customer if a vehicle has been assigned to it. Constraints (10) ensure the vehicle capacities are respected. Constraints (11) and (12) are degree constraints and subtour elimination constraints, respectively. Inequalities (13) ensure that at most one vehicle visits each customer in each period, thus forbidding split deliveries. Constraints (14)−(17) enforce integrality and non-negativity conditions on the variables.

This model can be strengthened through the inclusion of the following families of valid inequalities [6]:

$$x_{i0}^{kt} \leq 2y_{i}^{kt} \quad i \in \mathcal{V} \quad k \in \mathcal{K} \quad t \in \mathcal{T} \tag{18}$$

$$x_{ij}^{kt} \leq y_{i}^{kt} \quad i,j \in \mathcal{V} \quad k \in \mathcal{K} \quad t \in \mathcal{T} \tag{19}$$

$$y_{i}^{kt} \leq y_{0}^{kt} \quad i \in \mathcal{V}' \quad k \in \mathcal{K} \quad t \in \mathcal{T} \tag{20}$$

$$y_{0}^{kt} \leq y_{0}^{k-1,t} \quad k \in \mathcal{K}\backslash\{1\} \quad t \in \mathcal{T} \tag{21}$$

$$y_{i}^{kt} \leq \sum_{j<i} y_{j}^{k-1,t} \quad i \in \mathcal{V} \quad k \in \mathcal{K}\backslash\{1\} \quad t \in \mathcal{T} \tag{22}$$

Constraints (18) and (19) enforce the condition that if the supplier is the immediate successor of a customer in the route of vehicle k in period t, then i must be visited by the same vehicle. A similar reasoning is applied to customer j in inequalities (19). Constraints (20) ensure that the supplier is visited if any customer i is visited by vehicle k in period t.

When the vehicle fleet is homogeneous, one can break some of the vehicle symmetry by mean of constraints (21), thus ensuring that vehicle k cannot leave the depot if vehicle $k-1$ is not used. This symmetry breaking rule is then extended to the customer vertices by constraints (22) which state that if customer i is assigned to vehicle k in period t, then vehicle $k-1$ must serve a customer with an index smaller than i in the same period.

We also introduce additional cuts in order to strengthen this formulation. If the sum of the demands over $[t_1, t_2]$ is greater than or equal to the maximum possible inventory held, then there must be at least one visit to this customer in the interval $[t_1, t_2]$. This constraint can be strengthened by considering that if the quantity needed to satisfy future demands is larger than the maximum inventory capacity, then several visits are needed. Since the maximum delivery size is the minimum between the holding capacity and the maximum vehicle capacity, one can round up the right hand side of (23). Making the numerator tighter by considering the actual inventory instead of the maximum possible inventory yields inequalities (24), which cannot be rounded up because they would then become non-linear due to the presence of the $I_i^{t_1}$ variable in their right-hand side:

$$\sum_{k \in \mathcal{K}} \sum_{t'=t_1}^{t_2} y_i^{kt'} \geq \left\lceil \frac{\sum\limits_{t'=t_1}^{t_2} d_i^{t'} - C_i}{\min\{\max_k\{Q_k\}, C_i\}} \right\rceil \quad i \in \mathcal{V}' \quad t_1, t_2 \in \mathcal{T}, t_2 \geq t_1 \tag{23}$$

$$\sum_{k \in \mathcal{K}} \sum_{t'=t_1}^{t_2} y_i^{kt'} \geq \frac{\sum\limits_{t'=t_1}^{t_2} d_i^{t'} - I_i^{t_1}}{\min\{\max_k\{Q_k\}, C_i\}} \quad i \in \mathcal{V}' \quad t_1, t_2 \in \mathcal{T}, t_2 \geq t_1. \tag{24}$$

A different version of the same inequalities can be written as follows. It is related to whether the inventory hold at each period is sufficient to fulfil future demands. In particular, if the inventory held in period t_1 by customer i is not sufficient to fulfil future demands, then a visit to this customer must take place in the interval $[t_1, t_2]$. This condition can be enforced by the following set of valid inequalities:

$$\sum_{k \in \mathcal{K}} \sum_{t'=t_1}^{t_2} y_i^{kt'} \geq \frac{\sum\limits_{t'=t_1}^{t_2} d_i^{t'} - I_i^{t_1}}{\sum\limits_{t'=t_1}^{t_2} d_i^{t'}} \quad i \in \mathcal{V}' \quad t_1, t_2 \in \mathcal{T}, t_2 \geq t_1. \tag{25}$$

It is relevant to note that this model distinguishes items of different ages through the use of index g. The variables have a meaning similar to those of the multi-product IRP [7]. In the case of a single perishable product, the model works as if products of different ages are different from each other (through their index), and have different profits, but contrary to what happens in the multi-product case, any of these products can be used to satisfy the same demand. Another particularity of this model is that at each period, an item transforms itself into another one through the process of aging. Thus, our problem shares some features of the multi-product problem [7], but it is structurally different from it.

4 Branch-and-Cut Algorithm

For very small instances sizes, the model presented in Section 3 can be fully described and all constraints and variables generated. It can then be solved by feeding it directly into an integer linear programming solver. However, for instances of realistic sizes, the number of subtour elimination constraints (12) is too large to allow full enumeration and these must be dynamically generated throughout the search process. The exact algorithm we present is then a branch-and-cut scheme in which subtour eliminations constraints are only generated and added into the program whenever they are found to be violated. It works as follows. At a generic node of the search tree, a linear program containing a subset of the subtour elimination constraints is solved, a search for violated inequalities is performed, and some of these are added to the current program which is then reoptimized. This process is reiterated until a feasible or dominated solution is reached, or until there are no more cuts to be added. At this point, branching on a fractional variable occurs. We provide a sketch of the branch-and-bound-and-cut scheme in Algorithm 1.

5 Computational Experiments

In order to evaluate the proposed algorithm, we have coded it in C++ and used IBM Concert Technology and CPLEX 12.5 running in parallel with 10 threads. All computations were executed on a grid of Intel Xeon™ processors running at 2.66 GHz with up to 48 GB of RAM installed per node, with the Scientific Linux 6.1 operating system.

We have created randomly generated instances to assess the performance of our algorithm on a wide range of situations. The details regarding the parameters used to generate the instances are described in Appendix A. We have generated a total of 60 different instances which vary in terms of the number of customers, periods, vehicles and maximum age of the product. In what follows we provide average statistics over five instances per combination. Detailed results are presented in Appendix B. These results along with the instances are available in the website `http://www.leandro-coelho.com/instances`.

We provide in Table 1 average computational results for these instances. We have allowed the algorithm to run for a maximum of two hours. When the time limit is reached, we report the best available lower and upper bound (solution value) and the optimality gap. We report the instance sizes as *(n-s-K-H)*, where n is the number of customers, s is the maximum age of the product, K is the number of vehicles, and H is the length of the planning horizon. The next columns report the average best solution value obtained, the average best bound, the average optimality gap, the number of instances out of the five that were solved to optimality, and the average running time in seconds.

These results clearly indicate that the performance of the algorithm is directly related to the number of customers and to the length of the planning horizon. For the instances with shorter

Algorithm 1 Branch-and-cut algorithm

At the root node of the search tree, generate and insert all valid inequalities into the program.
Subproblem solution. Solve the LP relaxation of the current node.
Termination check:
if there are no more nodes to evaluate **then**
Stop.
else
Select one node from the branch-and-cut tree.
end if
while the solution of the current LP relaxation contains subtours **do**
Identify connected components as in Padberg and Rinaldi [23].
Determine whether the component containing the supplier is weakly connected as in Gendreau et al. [13].
Add all violated subtour elimination constraints (12).
Subproblem solution. Solve the LP relaxation of the current node.
end while
if the solution of the current LP relaxation is integer **then**
Go to the termination check.
else
Branching: branch on one of the fractional variables.
Go to the termination check.
end if

Table 1: Summary of the computational results for the PIRP

Instance size (n-s-K-H)	Best known solution value	Best known lower bound	Gap (%)	# solved	Time (s)
PIRP-10-2-1-3	31009.58	31009.58	0.00	5/5	0.2
PIRP-10-3-1-6	60412.56	60412.56	0.00	5/5	0.4
PIRP-10-5-1-10	80552.30	80552.30	0.00	5/5	4.8
PIRP-20-2-2-3	61912.98	61912.98	0.00	5/5	0.4
PIRP-20-3-2-6	127157.20	127157.20	0.00	5/5	547.4
PIRP-20-5-2-10	180218.40	176862.20	2.03	0/5	7205.2
PIRP-30-2-2-3	96995.56	96995.56	0.00	5/5	4.2
PIRP-30-3-2-6	191237.60	190195.80	0.54	1/5	5800.0
PIRP-30-5-2-10	294052.00	289241.40	1.75	0/5	7209.6
PIRP-40-2-3-3	126066.80	126066.80	0.00	5/5	67.4
PIRP-40-3-3-6	250671.40	245361.00	2.22	0/5	7214.6
PIRP-50-2-3-3	173794.60	173794.60	0.00	5/5	368.0

planning horizons, the algorithm is always able to find optimal solutions within a few seconds of computational time. This remains true even when the number of customers and vehicles increases, e.g., all five instances with 50 customers and three vehicles were solved to optimality, taking on average six minutes.

6 Conclusions

We have presented and discussed general assumptions for the joint replenishment and inventory control of perishable products. We have modeled the problem under general assumptions as a MILP, and we have solved it exactly by branch-and-cut. Our model remains linear even when the product value decreases in a non-linear or even in a non-convex fashion over time. It keeps track of the number of items of each age, and considers different holding costs for products of different ages. The model optimally determines which items to sell at each period based on the trade-off between cost and revenue. The algorithm can effectively compute optimal joint replenishment and delivery decisions for perishable products in an inventory-routing context for medium size instances. Computational experiments carried out on randomly generated instances support this conclusion.

Appendices

A Details of the instances

Our testbed is composed of instances generated with the following parameters:

- Number of customer n: 10, 20, 30, 40, 50;
- Number of periods H: 3 for up to $n = 50$; 6 for up to $n = 40$; and 10 for up to $n = 30$;

- Number of vehicles K: 1 for $n = 10$; 2 for $n = 20$ and 30; 3 for $n = 40$ and 50;
- Maximum age of the products s: 2 for $H = 3$; 3 for $H = 6$; 5 for $H = 10$;
- Demand d_i^t: randomly selected from the interval [30, 210];
- Position (x, y) of the supplier and customers: randomly selected from the interval [0, 1000];
- Customers' inventory capacity C_i: $R \times \max_t\{d_i^t\}$, where R is randomly selected from the set {2, 3};
- Initial inventory I_i^0 of fresh products: equal to $C_i - d_i^1$;
- Revenue u_i^g: equal to $R_1 - (R_1 - R_2)\, g/s$, where R_1 and R_2 are randomly selected from the intervals [10, 20] and [4, 7], respectively;
- Inventory holding cost h_i^g: equal to $\left(R_1 + gR_2/\left(1+g\right)\right)/100$, where R_1 and R_2 are randomly selected from the intervals [0, 100] and [0, 70], respectively;
- Vehicle capacities Q_k: equal to $\lfloor 1.25 \sum\limits_{i\in\mathcal{V}'} \sum\limits_{t\in\mathcal{T}} d_i^t/(HK)\rfloor$;

For each combination of the parameters above we have generated five instances, yielding a total of 60 instances.

B Detailed computational results

We present in Table 2 the detailed computational results for all instances.

References

[1] V. Aelion, F. Castells, and A. Veroutis. Life cycle inventory analysis of chemical processes. *Environmental Progress*, 14(3):193–200, 2006.

[2] H. Andersson, A. Hoff, M. Christiansen, G. Hasle, and A. Løkketangen. Industrial aspects and literature survey: Combined inventory management and routing. *Computers & Operations Research*, 37(9):1515–1536, 2010.

Table 2: Detailed results of the computational experiments for the PIRP

Instance size (n-s-K-H)	Best known solution value	Best known lower bound	Gap (%)	Time (s)
PIRP-10-2-1-3-1	28401.20	28401.20	0.00	1
PIRP-10-2-1-3-2	33547.80	33547.80	0.00	0
PIRP-10-2-1-3-3	27269.60	27269.60	0.00	0
PIRP-10-2-1-3-4	33106.90	33106.90	0.00	0
PIRP-10-2-1-3-5	32722.40	32722.40	0.00	0
PIRP-10-3-1-6-1	67252.10	67252.10	0.00	0
PIRP-10-3-1-6-2	49205.40	49205.40	0.00	1
PIRP-10-3-1-6-3	67453.00	67453.00	0.00	1
PIRP-10-3-1-6-4	66474.60	66474.60	0.00	0
PIRP-10-3-1-6-5	51677.70	51677.70	0.00	0
PIRP-10-5-1-10-1	81005.30	81005.30	0.00	7
PIRP-10-5-1-10-2	69173.70	69173.70	0.00	4
PIRP-10-5-1-10-3	101746.00	101746.00	0.00	3
PIRP-10-5-1-10-4	82240.30	82240.30	0.00	2
PIRP-10-5-1-10-5	68596.20	68596.20	0.00	8
PIRP-20-2-2-3-1	60857.70	60857.70	0.00	1
PIRP-20-2-2-3-2	75348.70	75348.70	0.00	0
PIRP-20-2-2-3-3	69482.60	69482.60	0.00	0
PIRP-20-2-2-3-4	52656.40	52656.40	0.00	0
PIRP-20-2-2-3-5	51219.50	51219.50	0.00	1
PIRP-20-3-2-6-1	109618.00	109618.00	0.00	226
PIRP-20-3-2-6-2	135452.00	135452.00	0.00	38
PIRP-20-3-2-6-3	107555.00	107555.00	0.00	2041
PIRP-20-3-2-6-4	133538.00	133538.00	0.00	403
PIRP-20-3-2-6-5	149623.00	149623.00	0.00	29
PIRP-20-5-2-10-1	197453.00	194680.00	1.42	7205
PIRP-20-5-2-10-2	156088.00	151469.00	3.04	7207
PIRP-20-5-2-10-3	182977.00	180894.00	1.15	7205
PIRP-20-5-2-10-4	148117.00	143299.00	3.36	7205
PIRP-20-5-2-10-5	216457.00	213969.00	1.16	7204
PIRP-30-2-2-3-1	82232.30	82232.30	0.00	2
PIRP-30-2-2-3-2	93193.20	93193.20	0.00	2
PIRP-30-2-2-3-3	100433.00	100433.00	0.00	7
PIRP-30-2-2-3-4	115649.00	115649.00	0.00	8
PIRP-30-2-2-3-5	93470.30	93470.30	0.00	2
PIRP-30-3-2-6-1	188559.00	186055.00	1.34	7207
PIRP-30-3-2-6-2	196276.00	196276.00	0.00	174
PIRP-30-3-2-6-3	179968.00	179529.00	0.24	7205
PIRP-30-3-2-6-4	175391.00	174763.00	0.35	7206
PIRP-30-3-2-6-5	215994.00	214356.00	0.76	7208
PIRP-30-5-2-10-1	229619.00	223817.00	2.59	7211
PIRP-30-5-2-10-2	259054.00	254304.00	1.86	7209
PIRP-30-5-2-10-3	317954.00	313020.00	1.57	7209
PIRP-30-5-2-10-4	371578.00	368057.00	0.95	7211
PIRP-30-5-2-10-5	292055.00	287009.00	1.75	7208
PIRP-40-2-3-3-1	135053.00	135053.00	0.00	36
PIRP-40-2-3-3-2	129153.00	129153.00	0.00	50
PIRP-40-2-3-3-3	123732.00	123732.00	0.00	101
PIRP-40-2-3-3-4	117908.00	117908.00	0.00	88
PIRP-40-2-3-3-5	124488.00	124488.00	0.00	62
PIRP-40-3-3-6-1	236926.00	228613.00	3.63	7215
PIRP-40-3-3-6-2	222670.00	218322.00	1.99	7219
PIRP-40-3-3-6-3	258265.00	252088.00	2.45	7217
PIRP-40-3-3-6-4	250888.00	244936.00	2.43	7206
PIRP-40-3-3-6-5	284608.00	282846.00	0.62	7216
PIRP-50-2-3-3-1	165698.00	165698.00	0.00	87
PIRP-50-2-3-3-2	184855.00	184855.00	0.00	743
PIRP-50-2-3-3-3	175569.00	175569.00	0.00	35
PIRP-50-2-3-3-4	165574.00	165574.00	0.00	891
PIRP-50-2-3-3-5	177277.00	177277.00	0.00	84

[3] S. Axsäter. *Inventory control*. International Series in Operations Research & Management Science, volume 90. Springer, New York, 2006.

[4] J. Blackburn and G. Scudder. Supply chain strategies for perishable products: The case of fresh produce. *Production and Operations Management*, 18(2):129–137, 2009.

[5] E. Brodheim, C. Derman, and G. P. Prastacos. On the evaluation of a class of inventory policies for perishable products such as blood. *Management Science*, 21(11):1320–1325, 1975.

[6] L. C. Coelho and G. Laporte. The exact solution of several classes of inventory-routing problems. *Computers & Operations Research*, 40(2):558–565, 2013.

[7] L. C. Coelho and G. Laporte. A branch-and-cut algorithm for the multi-product multi-vehicle inventory-routing problem. *International Journal of Production Research*, Forthcoming, 2013. doi: 10.1080/00207543.2012.757668.

[8] L. C. Coelho, J.-F. Cordeau, and G. Laporte. Thirty years of inventory-routing. *Transportation Science*, Forthcoming, 2013.

[9] P. D. Cumming, K. E. Kendall, C. C. Pegels, J. P. Seagle, and J. F. Shubsda. A collections planning model for regional blood suppliers: description and validation. *Management Science*, 22(9):962–971, 1976.

[10] A. Custódio and R. Oliveira. Redesigning distribution operations: a case study on integrating inventory management and vehicle routes design. *International Journal of Logistics*, 9(2):169–187, 2006.

[11] H. Emmons. A replenishment model for radioactive nuclide generators. *Management Science*, 14 (5):263–274, 1968.

[12] P. Farahani, M. Grunow, and H.-O. Günther. Integrated production and distribution planning for perishable food products. *Flexible Services and Manufacturing Journal*, 24(1):28–51, 2012.

[13] M. Gendreau, G. Laporte, and F. Semet. The covering tour problem. *Operations Research*, 45(4): 568–576, 1997.

[14] V. Hemmelmayr, K. F. Doerner, R. F. Hartl, and M. W. P. Savelsbergh. Vendor managed inventory for environments with stochastic product usage. *European Journal of Operational Research*, 202(3): 686–695, 2010.

[15] R. Jagannathan and T. Sen. Storing crossmatched blood: a perishable inventory model with prior allocation. *Management science*, 37(3):251–266, 1991.

[16] J. B. Jennings. Blood bank inventory control. *Management Science*, 19(6):637–645, 1973.

[17] K. E. Kendall and S. M. Lee. Formulating blood rotation policies with multiple objectives. *Management Science*, 26(11):1145–1157, 1980.

[18] G. Laporte. Fifty years of vehicle routing. *Transportation Science*, 43(4):408–416, 2009.

[19] T. Le, A. Diabat, J.-P. Richard, and Y. Yih. A column generation-based heuristic algorithm for an inventory routing problem with perishable goods. *Optimization Letters*, 2012. doi: 10.1007/s11590-012-0540-2.

[20] A. Mercer and X. Tao. Alternative inventory and distribution policies of a food manufacturer. *Journal of the Operational Research Society*, 47(6):755–765, 1996.

[21] S. Nahmias. *Perishable Inventory Systems*. International Series in Operations Research & Management Science, volume 160. Springer, New York, 2011.

[22] A. Osvald and L. Z. Stirn. A vehicle routing algorithm for the distribution of fresh vegetables and similar perishable food. *Journal of Food Engineering*, 85(2):285–295, 2008.

[23] M. W. Padberg and G. Rinaldi. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM Review*, 33(1):60–100, 1991.

[24] C. C. Pegels and A. E. Jelmert. An evaluation of blood-inventory policies: a Markov chain application. *Operations Research*, 18(6):1087–1098, 1970.

[25] S. P. Singh, V. Chonhenchob, and J. Singh. Life cycle inventory and analysis of re-usable plastic containers and display-ready corrugated containers used for packaging fresh fruits and vegetables. *Packaging Technology and Science*, 19(5):279–293, 2006.

[26] K. van Donselaar, T. van Woensel, R. A. C. M. Broekmeulen, and J. Fransoo. Inventory control of perishables in supermarkets. *International Journal of Production Economics*, 104(2):462–472, 2006.

[27] X. Yang. *Choosing transportation alternatives for highly perishable goods: a case study on nuclear medicine*. PhD thesis, Massachusetts Institute of Technology, 2006.