European J. of Industrial Engineering

ISSN online: 1751-5262 - ISSN print: 1751-5254
https://www.inderscience.com/ejie

The influence of shelf life on the integrated production scheduling and vehicle routing optimisation for perishable products

Hercules Tadeu Asato Dantas, Roberto Fernandes Tavares Neto, Juliana Keiko Sagawa

DOI: 10.1504/EJIE.2024.10058090

Article History:

Received:	26 July 2022
Last revised:	07 June 2023
Accepted:	07 June 2023
Published online:	01 October 2024

The influence of shelf life on the integrated production scheduling and vehicle routing optimisation for perishable products

Hercules Tadeu Asato Dantas*,
Roberto Fernandes Tavares Neto and
Juliana Keiko Sagawa

Departamento de Engenharia de Produção,
Universidade Federal de São Carlos – UFSCar,
São Carlos, SP, Brazil
Email: dantas.hercules@gmail.com
Email: tavares@dep.ufscar.br
Email: juliana@dep.ufscar.br
*Corresponding author

Abstract: Prior studies on integrating production and distribution optimisation highlight the benefits of such an approach. However, no studies regarding standardised metrics for categorising shelf life lengths in different scenarios were found. Moreover, the current literature does not address the change of behaviour on solving strategies accordingly to different shelf life profiles. To address these gaps, this research proposes the normalised shelf life metric to classify long and short shelf life products. Additionally, we present a simplified optimisation formulation for the integrated production scheduling and vehicle routing problem. By evaluating three solution methods (MILP model, genetic algorithm, and logic-based Benders decomposition model), our findings reveal that shelf life significantly affects solution performance. These results emphasise the need for research focused on solution methods tailored specifically for short shelf life products. [Received: 26 July 2022; Accepted: 7 June 2023]

Keywords: shelf life; production-distribution integration; perishable products; genetic algorithm.

Reference to this paper should be made as follows: Dantas, H.T.A., Tavares Neto, R.F. and Sagawa, J.K. (2024) 'The influence of shelf life on the integrated production scheduling and vehicle routing optimisation for perishable products', *European J. Industrial Engineering*, Vol. 18, No. 6, pp.860–884.

Biographical notes: Hercules Tadeu Asato Dantas holds an MSc in Industrial Engineering from Federal University of São Carlos, and a Bachelor's in Industrial Engineering from Federal University of São Carlos. His research interests include optimisation methods applied to logistics problems.

Roberto Fernandes Tavares Neto is an Associated Professor at the Federal University of Sao Carlos. He holds a PhD and MSc in Industrial Engineering, focusing on the application of artificial intelligence to solve logistics and

scheduling problems. He was a Visiting Scholar at the University of Graz and Université Libre de Bruxelles, enhancing his focus on integrated production planning and control and algorithm design. Seeking a high integration between academy and industry, he has been developing joint works with several companies, approaching problems ranging from those from the oil industry, logistics, maintenance, and manufacturing production planning and scheduling.

Juliana Keiko Sagawa is an Associate Professor at the Department of Production Engineering of the Federal University of São Carlos, Brazil. She holds a PhD in Production Engineering from the University of São Paulo (USP) and had a temporary research stay at Universität Bremen, Germany. Her research interests encompass dynamic modelling and control theory applications to manufacturing, intelligent production systems, production planning and control systems, strategy and organisational integration.

1 Introduction

There are several reasons why the integrated approach for production and distribution optimisation is becoming more popular. Computational studies show that this approach can lead to cost reductions ranged from 3% to 20% when compared with a sequential planning (Chandra and Fisher, 1994), i.e., when the production schedule is input for the distribution schedule. Besides the benefits in costs, situations where the inventory is constrained also benefit from adopting an integrated approach. Examples of that are found in companies that adopt the make-to-order production system (Chen, 2010) or companies that handle perishable products (Marandi and Zegordi, 2017), which are goods that have physical deterioration or a decrease in the customer's value perception over time (Amorim et al., 2013b). In this case, inefficiencies in the production planning will affect the distribution system. Therefore, several authors often highlight the attractiveness of the integrated planning (Amorim et al., 2012; Chen, 2010).

One characteristic that significantly influences operational planning for a perishable product is its shelf life. When analysing the integration of production scheduling and vehicle routing problem for a product with a fixed shelf life, Amorim et al. (2012) and Farahani et al. (2012) verified that the integrated approach value is leveraged for products with a higher perishability degree. Despite these encouraging results, there is little knowledge regarding how the problem's characteristics impact the integration of those two sub-problems (Moons et al., 2017). This paper will refer to that integrated problem as integrated production and distribution scheduling problem for perishable products (IPDSP-P).

Both the production scheduling and the vehicle routing problem are usually classified as NP-Hard. Thus, Section 5 an investigation on the performance of commercial solvers for the IPDSP-P. Furthermore, we also investigate how the shelf life influences the expected of several solution methods. In this case, a good expected performance would be the solution method that could find solutions for the IPDSP-P in short time.

A better understanding of how the shelf life impacts the expected performance of the solution methods to solve the IPDSP-P may lead to a better choice in a project that aims to solve this kind of problem. For example, if a shorter shelf life does make the IPDSP-P harder to solve, a standard MILP model implemented in a commercial solver may be inappropriate. Thus, besides the MILP model, we evaluate two other solution approaches: an exact method using logic-based Benders decomposition (LBBD) and a genetic algorithm (GA). The Genetic Algorithm was chosen due to the ease of integrating additional techniques into the steps of the algorithm and the promising results found in different applications (e.g., see Shin et al., 2022; Di Placido et al., 2022; Türkyılmaz et al., 2022). The other solution approach analysed, the Logic-Based Benders Decomposition, was selected because one major challenge of solving an integrated problem using an exact method is the size of the resulting model (Grossmann, 2005; Papageorgiou, 2009; Garcia and You, 2015). Thus, decomposition methods are often applied to address the integrated problems (Grossmann, 2012). Moreover, the structural properties of the problem (presented in Subsection 4.1) suggested that the IPDSP-P could be decomposed into a master problem and subproblems, which supports the choice of the LBBD as a solution approach.

But what could be considered a short shelf life? Looking in the literature, we found several different metrics to calculate a product shelf life. Therefore, we propose a standard metric called normalised shelf life and analysed several papers using this metric to be able to distinguish a short from a long shelf life.

Besides the comparison of the influence of the shelf life on IPDSP-P and the normalised shelf life, another contribution of this study includes a form to simplify the problem. We prove that, given a feasible route for the problem, if there is any production schedule that turns the IPDSP-P feasible, then the production schedule that follows the same sequence as the given route will also be feasible.

The remainder of this paper is organised as follows. Section 2 reviews the existing literature on production and distribution integration for perishable products. The mathematical formulation for IPDSP-P is provided in Section 3. In Section 4, we present our approach to solving the IPDSP-P using the logic-based Benders decomposition and a genetic algorithm. Then, the proposed metric called normalised shelf life and the computational results are presented in Section 5. Finally, conclusions and further research opportunities are presented in Section 6.

2 Literature review

A significant set of researchers studied integrated production-distribution problems. For reviews on models that address the strategic level refer to Vidal and Goetschalckx (1997), for tactical level refer to Mula et al. (2010), and for operational level refer to Chen (2010), Moons et al. (2017) and Reimann et al. (2014). This section will focus on operational production-distribution planning specifically for perishable products.

Amorim et al. (2013a) developed two MIP models to solve the integrated production and distribution problem for a set of parallel lines with limited capacity to produce several perishable products, whose shelf life was 3 or 5 times the average production time of a demand order. Several homogeneous vehicles deliver the goods, respecting the routes stated by the model. The study's objective was to evaluate if lot-sizing has advantages over a situation that only considers the batching of orders.

Devapriya et al. (2017) opted for an evolutionary approach. They developed a genetic algorithm and two memetic algorithms to minimise the distribution cost by defining the production schedule and the vehicles' routes for a plant that produces a single product, whose shelf life is equal to 50. In this problem, the vehicles may start another trip after returning to the depot from a delivery.

In a similar problem, Marandi and Zegordi (2017) also created an evolutionary approach algorithm based on particle swarm optimisation, named as improved particle swarm optimisation. In this study, the product shelf life was randomly generated from a uniform distribution, whose lower and upper bound were calculated based on processing time and travel distance.

Although the vehicle routing problem is common in the integrated production-distribution planning literature, direct shipment is also popular. Li et al. (2016) proposed a MILP model to solve the production inventory routing planning. Seyedhosseini and Ghoreyshi (2014) consider a similar problem. To solve it, the authors employed an MILP modal and a particle swarm optimisation to solve the problem. In addition, they considered a shelf life varying between 2 and 3 periods.

More recently, Solina and Mirabelli (2021) applied an optimisation model to solve an integrated production and distribution problem for an Italian company that produces, stores, and distributes vegetables. The model minimises distribution, inventory, and energy consumption costs by deciding how much to produce, store, and deliver to customers each day. The perishability was covered by the raw materials that must be processed before their shelf life end. The number of vehicles is also limited.

Fikar et al. (2021) built a decision support system for e-grocery companies. That system has an agent-based simulation model, which is supported by heuristics. Those heuristics are responsible for the inventory policy and the customer's order routing. The shelf life is a stochastic variable that follows a normal distribution.

This brief review points out that studies on IPDSP-P use exact models, heuristics, and metaheuristics to solve the problem. Thus, understanding the influence of shelf life on the difficulty of solving the IPDSP-P will help choose a proper technique. We also observed a lack of a standard metric when selecting the parameters to distinguish between short or long shelf life in an operational planning context.

3 Problem statement and model formulations

In this section, we present a mathematical formulation for the IPDSP-P as presented by Dantas (2022). This problem consists of a single machine or plant, which produces a particular perishable product whose quality starts to decay right after the production starts. This product must be delivered to a set of $|N|$ customers. The delivery can occur only while the product still has some shelf life and there is available capacity in one of the $|K|$ identical fixed capacity vehicles. Following the common representations of this problem (e.g., see Dantzig and Ramser, 1959), those vehicles must be routed through a set A of arcs contained in a directed graph G = (V,A). The clients and the plant represent the vertices of this graph (V), $V = N \cup \{0, n+1\}$. Two vertices 0 and $n+1$ represent the plant (i.e., initial and final points of the route). Figure 1 illustrates the problem.

The notation adopted for this IPDSP-P model is:

- *Indices*

  $(c, d) \in N$ customers

  $k \in K$ vehicles.

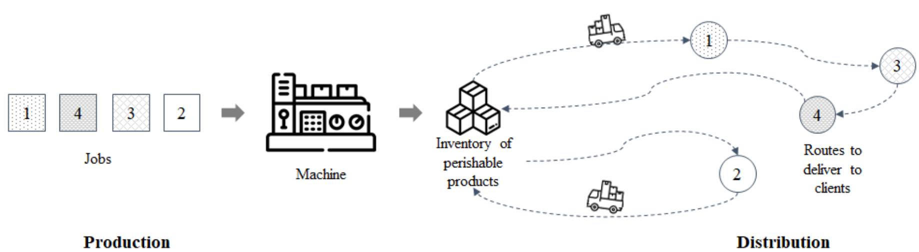
- *Parameters*

dem_c	demand at customer c (units)
p_i	processing time for the customer i order
$CapV$	vehicle capacity
ct_{cd}	cost to transport products from customer c to d
tt_{cd}	travel time to transport products from customer c to d
$\overline{ft}$	fixed cost associated to each vehicle k
sl	product's shelf life.

- *Variables*

x_{cd}^k	equals to 1, if vehicle k depart from customer c to customer d (0 otherwise)
w_c^k	starting time at which customer c is serviced by vehicle k
V_{cd}	equals to 1, if customer d order is produced right after customer c order (0 otherwise)
Ct_c	completion time for customer c order.

Figure 1 A graphical example of IPDSP-P



Based on those elements, the IPDSP-P may be formulated as follows:

$$Min\ \overline{ft} \sum_{k \in K} (1 - x_{0,n+1}^k) + \sum_{k \in K} \sum_{c,d \in N} ct_{cd} x_{cd}^k \tag{1}$$

Subject to

$$Ct_0 = 0 \tag{2}$$

$$\sum_{c \in N} V_{cd} = 1 \qquad \forall d \in N \tag{3}$$

$$\sum_{d \in N} V_{cd} = 1 \qquad \forall c \in N \tag{4}$$

$$Ct_d \geq Ct_c + p_d - M(1 - V_{cd}) \qquad \forall \begin{cases} d \in N \\ c \in N/\{0, d\} \end{cases} \tag{5}$$

$$\sum_{k\in K}\sum_{d\in N} x_{cd}^k = 1 \qquad \forall c \in N \tag{6}$$

$$\sum_{d\in N} x_{0d}^k = 1 \qquad \forall k \in K \tag{7}$$

$$\sum_{c\in N} x_{cd}^k - \sum_{c\in N} x_{dc}^k = 0 \qquad \forall k \in K; d \in N \tag{8}$$

$$\sum_{c\in N} x_{c,n+1}^k = 1 \qquad \forall k \in K \tag{9}$$

$$w_d^k \geq w_c^k + tt_{cd} - M(1 - x_{cd}^k) \qquad \forall k \in K; c, d \in N \tag{10}$$

$$\sum_{c\in N} dem_c \sum_{d\in N} x_{cd}^k \leq CapV \qquad \forall k \in K; c \in N \tag{11}$$

$$Ct_c - p_c + sl - \sum_{k\in K} w_c^k \geq 0 \qquad \forall c \in N \tag{12}$$

$$w_0^k \geq Ct_c - M\left(1 - \sum_{d\in N} x_{cd}^k\right) \qquad \forall k \in K; c \in N \tag{13}$$

$$w_d^k, Ct_c \geq 0; x_{cd}^k, V_{cd} \in \{0, 1\} \tag{14}$$

The objective function (1) minimises the distribution costs, which are composed of variable and fixed costs, based on total travel time and the number of vehicles used, respectively. The objective function does not consider production costs.

Constraints (2) to (4) are used to set the production sequence of customers' orders and constraint (5) establish when each order is completed.

Constraints (6) to (11) refer to distribution process. Constraint (6) ensures that each origin has only one destination and is visited only by one vehicle. Constraints (7) and (9) establish that each vehicle departing from the plant has only one destination and each vehicle returning to the plant has only one origin, respectively. Constraint (8) ensures that if a vehicle visits a client c, this node will be the next origin. Constraint (10) establishes the time when a vehicle that departs from c to d will arrive at node d. Constraint (11) enforces that vehicle capacity is respected. As stated in Amorim et al. (2013a), $x_{0,n+1}^k = 1$ means that the vehicle was not used.

Finally, constraint (12) enforces the product is delivered while it still has some shelf life, constraint (13) links the production to the distribution problem. The domain of variables is stated in equation (14).

4 Solution approach

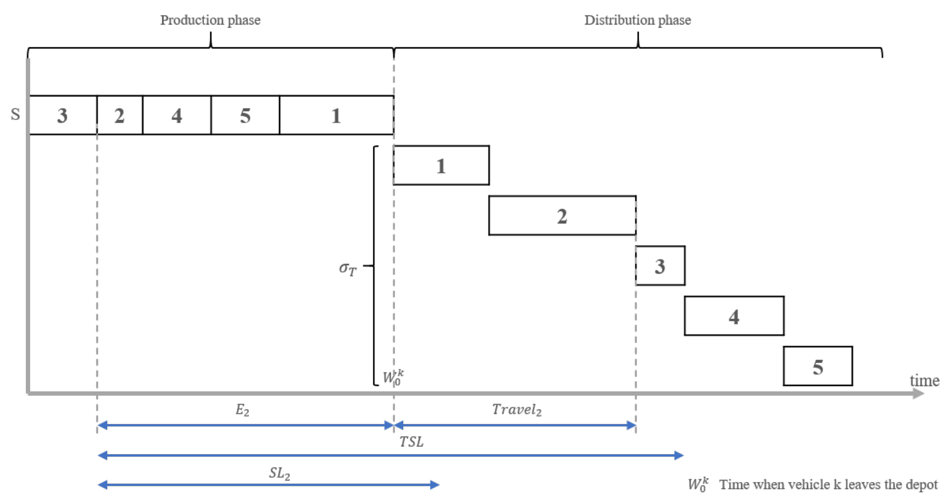
4.1 Simplifying the IPDSP-P

In this section, structural properties, used subsequently to simplify the model formulation of the IPDSP-P are analysed. We found that if any production sequence is feasible for a given distribution route, then the production sequence that has the same order as the distribution route is also feasible. As the production schedule does not impact the objective function, our goal is solely find a production schedule that satisfies the shelf life constraints for the optimal distribution route. This simplification is possible if the following assumptions are true:

- single machine environment
- the lot-size for each job is given
- single product with a single shelf life length
- no sequence dependent setup time or costs
- the product's quality starts to decay at the starting time of the production.

Lemma 1: Let $\sigma_k \subset A$ be a feasible route for of vehicle k for the distribution part of IPDSP-P. The route for vehicle k will be feasible only if $E_i \le SL_i \quad \forall i \in |\sigma_k|$, where $SL_i = TSL - Travel_i$. E_i is the time between the start of the production of order i and its departure from the plant, TSL is the total shelf life of the perishable product, and $Travel_i$ is the time elapsed between the departure from the plant and the delivery to the customer for the given route.

Figure 2 A graphic representation of the theorem's variables and parameters (see online version for colours)



Proof: The route σ_k will IPDSP-P will be feasible only if the total elapsed time between production and delivery for every customer order is lower than the total shelf life, i.e.,

$$E_i + Travel_i \le TSL \quad \forall i = 1, 2, ..., |\sigma_k| \tag{15}$$

Rearranging equation (15), we prove that IPDSP-P will be feasible only if $E_i \le SL_i$.
Figure 2 presents a graphical illustration of variables $Travel_i$, E_i, SL_i and TSL.□

Lemma 2: If the route $\sigma_k \subset A$ and the production sequence $S = \{x_1, x_2, ..., x_{|\sigma_k|}\}$ provide a feasible solution for IPDSP-P, then $E_i \le SL_i$ for the jobs $i = 1, 2, ..., x_1$ in sequence $S' = \{1, ..., x_1, ..., |\sigma_k|\}$

Proof: As the route is given and follows the order 1, 2, ..., $|\sigma_k|$, we may conclude that:

$$Travel_1 \leq Travel_2 \leq ... \leq Travel_{|\sigma_k|} \tag{16}$$

Since TSL is constant, the following relationship can be stated:

$$SL_{|\sigma_k|} \leq SL_{|\sigma_k|-1} \leq ... \leq SL_1 \tag{17}$$

Denoting Eo_i and E'_i as the E_i variable for S and S', respectively, and as x_1 is the first job of the sequence S, and S is feasible, we may assure that:

$$Eo_{x_1} = \sum_{i=1}^{|\sigma_k|} p_i \leq SL_{x_1} \leq SL_{x_1-1} \leq SL_{x_1-2} \leq ... \leq SL_1 \tag{18}$$

where p_i is the processing time of any job i.

As S and S' have the same number of jobs, and job 1 is the first job of sequence S', we may affirm that $E'_1 = Eo_{x_1}$, and for any other job in S', the following is true:

$$E'_j = E'_{j-1} - p_i \quad \forall j > 1 \tag{19}$$

As $p_i \geq 0$ then:

$$E'_{|\sigma_k|} \leq E'_{x_1} \leq E'_{x_1-1} \leq ... \leq E'_1 = Eo_{x_1} = \sum_{i=1}^{|\sigma_k|} p_i \tag{20}$$

Figure 3 A graphic representation of Lemma 2 (see online version for colours)

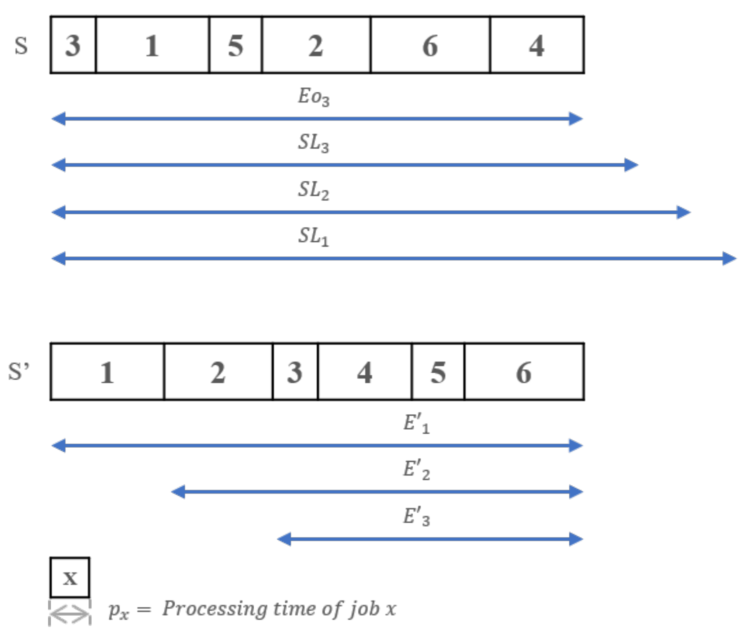
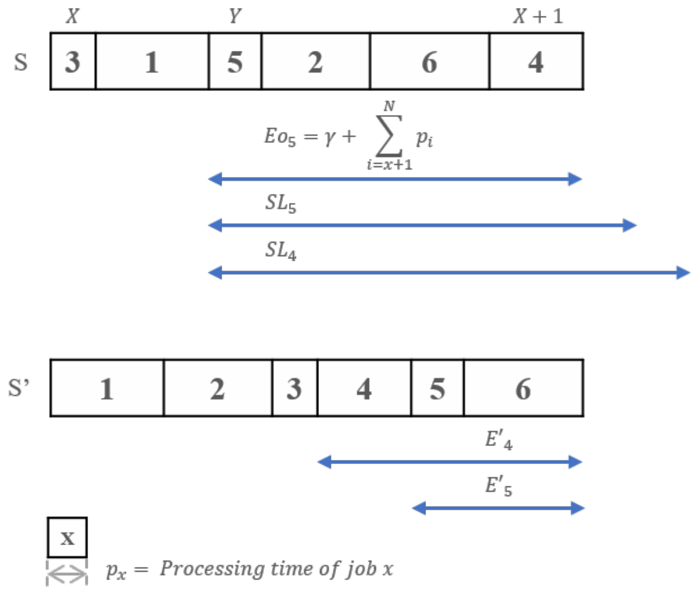


Figure 4 A graphic representation of Lemma 3 (see online version for colours)



According to (18) and (20):

$$E'_{|\sigma_k|} \leq E'_{x_1} \leq E'_{x_1-1} \leq ... \leq E'_1 = Eo_{x_1} \leq SL_{x_1} \leq SL_{x_1-1} \leq ... \leq SL_1 \quad (21)$$

Therefore,

$$E'_i \leq SL_i \quad \forall i \leq x_1 \quad (22)$$

Figure 3 presents a graphical illustration of the lemma. □

Lemma 3: If the route $\sigma_t = \{1, 2, ..., |\sigma_k|\}$ and the production sequence $S = \{x_1, x_2, ..., x_{|\sigma_k|}\}$ provide a feasible solution for IPDSP-P. Then, $E_i \leq SL_i$ for any job i that $x_1 + 1 \leq i \leq y$ in sequence $S' = \{1, ..., x_1, ..., y, ..., |\sigma_k|\}$, where y is the next job in S that $y > x_1$.

Proof: Since $y > x_1$, there will exist no job y' scheduled after x_1 and before y that $y' > x_1$. Thus:

$$Eo_y = \gamma + \sum_{i=x_1+1}^{|\sigma_k|} p_i \leq SL_y \quad (23)$$

Also,

$$E'_{x_1+1} = \sum_{i=x_1+1}^{|\sigma_k|} p_i \leq SL_{x_1+1} \quad (24)$$

γ is a real number ≥ 0 and, from equations (23) and (24), $E'_{x_1+1} \leq Eo_y$. Figure 4 provides a graphical support to prove that (24) is true. Finally, by the same logic presented in (21):

$$E'_{|\sigma_k|} \leq E'_y \leq E'_{y-1} \leq E'_{x_1+1} \leq Eo_y \leq SL_y \leq SL_{y-1} \leq ... \leq SL_{x_1+1} \tag{25}$$

Therefore,

$$E'_i \leq SL_i \quad \forall \quad x_1 + 1 \leq i \leq y \tag{26}$$

□

Theorem 1: If the route $\sigma_k = \{1, 2, ..., |\sigma_k|\}$ and the scheduling sequence $S = \{x_1, x_2, ..., x_{|\sigma_k|}\}$ provide a feasible solution for IPDSP-P, then the route σ_k and the production sequence $S' = \{1, 2, ..., |\sigma_k|\}$ will also be feasible.

Proof: Lemma 2 proves that the production sequence S' is feasible for all jobs $i < x_1$. Then, by repeating Lemma 3 until $y = |\sigma_k|$, the theorem is proved. □

The following corollary then follows immediately.

Corollary 1: If the route $\sigma_k = \{1, 2, ..., |\sigma_k|\}$ and the scheduling sequence $S = \{x_1, x_2, ..., x_{|\sigma_k|}\}$ provide the optimal solution for IPDSP-P, then the route σ_k and the production sequence $S' = \{1, 2, ..., |\sigma_k|\}$ will also be optimal.

Proof: As the production sequence does not directly influence the objective function of IPDSP-P, the objective function will be the same for S and S'. Thus, S' is optimal. □

4.2 Logic-based Benders decomposition approach to solve the IPDSP-P

From the theorem presented in the previous subsection, we found an opportunity to create a model based on the decomposition of the integrated problem. For this reason, we came up with a model using the logic-based Benders decomposition (LBBD). Our expectation for this model is that it can solve larger instances of IPDSP-P in comparison to the MILP model.

As presented in Hooker and Ottosson (2003), the LBBD approach extends Benders decomposition strategy of 'learning from one's mistake' to a broader class of problems. This approach partitions the problem into a master problem and one or more subproblems. The subproblems are easier to solve, and their solution provides information to the master problem, making it easier to solve.

To apply the LBBD to the IPDSP-P, we consider as the master problem a relaxed version of the model presented in Section 3, dropping all constraints related to the production scheduling (2)–(5), and solve it using a commercial MILP solver. When a solution is found, we evaluate if the IPDSP-P is feasible, considering the same production schedule as the vehicles' routes, which is based on the theorem presented in Subsection 4.1. If the evaluation turns that the solution (set T) is infeasible, we add cuts, presented in equation (27), that forbids any infeasible sub-tour (set $NT \subset T$) and the solution procedure of the master problem resumes. Algorithm 1 presents the described process.

$$\sum x_{ijk} < |NT| \quad \forall \quad k \in K \tag{27}$$

Algorithm 1 Logic-based Benders decomposition for IPDSP-P

repeat
    Run a MILP model considering just the distribution part (VRP problem) of IPDSP-P until a solution is found
    if solution is found **then**
        for all route in solution **do**
            ProductionScheduling = route
            for all customer in route **do**
                TotalTime ← Time elapsed between the start of the production and the delivery of the customer order
                if TotalTime > Available shelf life **then**
                    Create a constraint to eliminate the route
                end if
            end for
        end for
    end if
until Solution - LowerBound ≤ Maximum GAP
return OptimalSolution

According to Chu and Xia (2004), our algorithm will finitely converge to the optimality of the original problem if the cut we introduce satisfies two conditions:

1 if an infeasible solution is found in the master problem, the cut must exclude at least that solution

2 the cut must not exclude any feasible solution for the IPDSP-P.

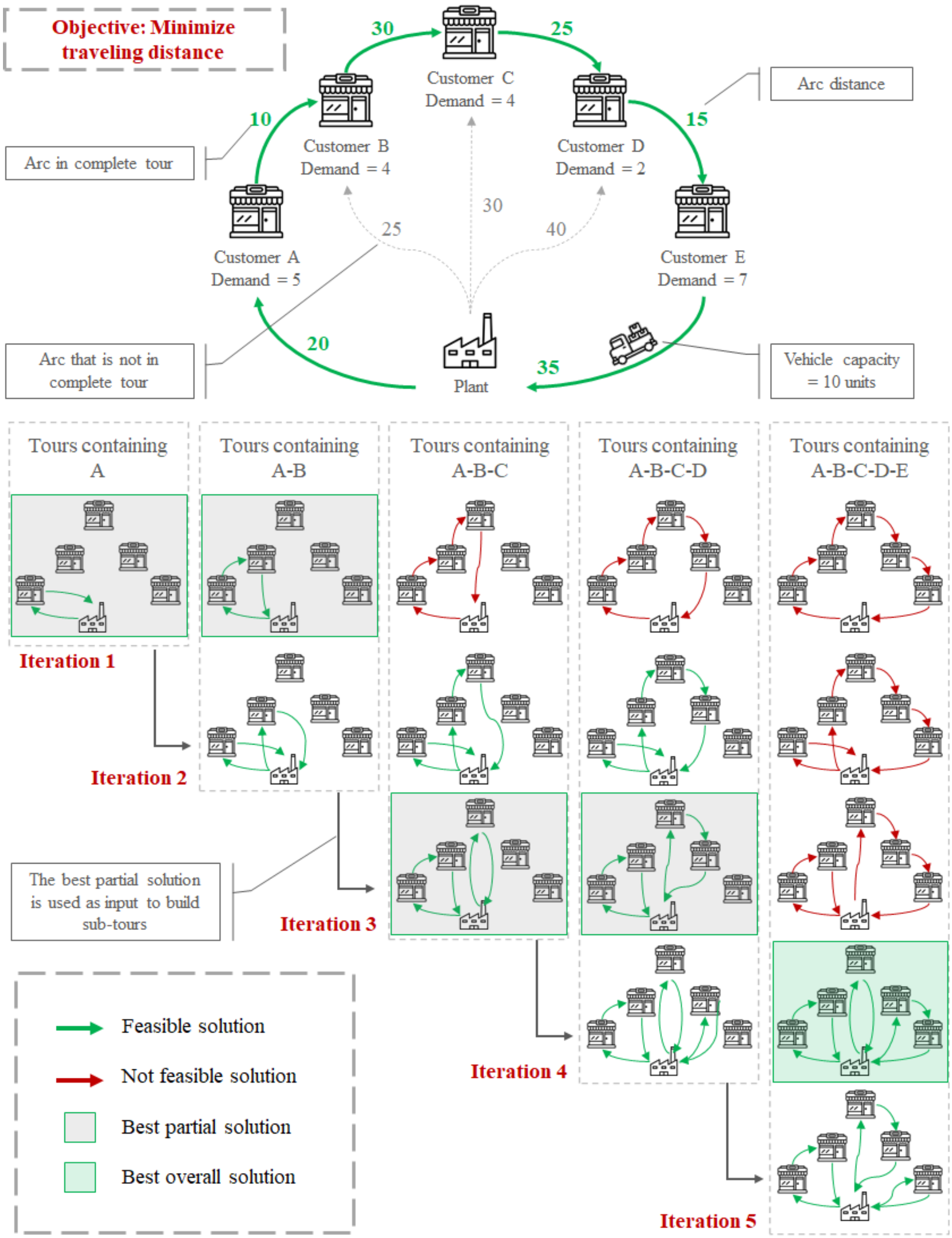
While we do exclude the infeasible route, our cut will not exclude any feasible solution. It is true because, based on the theorem in Subsection 4.1, it is impossible to exist any other production schedule that turns the IPDSP-P feasible when the schedule that uses the same sequence of the vehicles' routes turns the IPDSP-P infeasible.

4.3 Evolutionary approach to solve IPDSP-P

As seen in the literature review, evolutionary approaches are popular methods to solve the IPDSP-P because they can obtain high-quality solutions in a short amount of computational time (Devapriya et al., 2017). Thus, we also propose a genetic algorithm, taking into account the theorem presented in Subsection 4.1 to support our study.

To tackle the obstacles created by both the vehicle's capacity and the product's shelf life, we included the Split algorithm to produce feasible solutions for the problem.

Figure 5 A graphical example of split algorithm (see online version for colours)



Note: Icons made by Monkik and Surang from flaticon.com.

4.3.1 Split algorithm for IPDSP-P

The split algorithm was introduced by Prins (2004). As the problem consists in allocating and creating routes to serve customers using several capacitated vehicles, the algorithm choose the solution with the best overall result after building several options of feasible sub-tours, i.e., sub-tours that satisfies both vehicle capacity and shelf life constraints. In this process, the number of vehicles, the vehicles' routes and how customers will be served are the decision variables of the problem. Figure 5 illustrates this process.

Figure 6 A graphical example of one chromosome and its solution for a problem with nine customers/orders

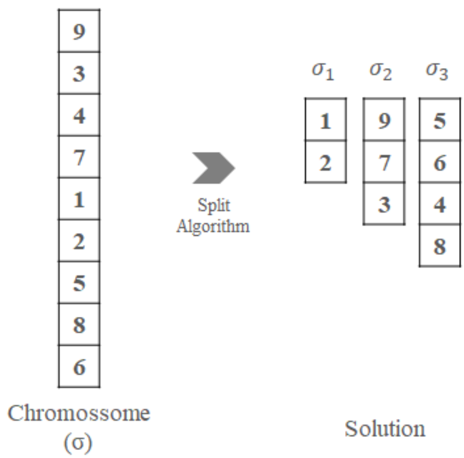
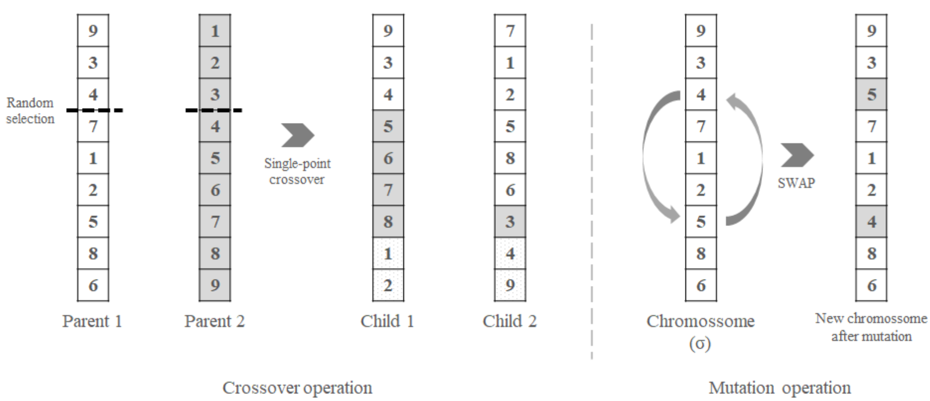


Figure 7 Illustration of the crossover and the mutation operations used in this study



Referring to the production component of the problem, we propose that customers' orders must be scheduled in the same order as sub-tours, based on the theorem presented in Subsection 4.1. For example, consider a IPDSP-P with three vehicles $K = (k_1, k_2, k_3)$, and each vehicle k depart from depot at time W_0^k, where $W_0^{k_1} < W_0^{k_2} < W_0^{k_3}$. If vehicle k_1 serves customers a, b and c in that order, i.e., $\sigma_{k_1} = (a, b, c)$;

and $\sigma_{k_2} = (d)$ and $\sigma_{k_3} = (e, f)$. Then, production schedule sequence will be $seq = (a, b, c, d, e, f)$.

4.3.2 Genetic algorithm for IPDSP-P

The genetic algorithm is a family of models based on the population evolution process. Its implementation starts with a population of random chromosomes. Those chromosomes are data structures that represent a solution to the target problem. The evolution process occurs by combining those chromosomes to produce new (and possibly better) solutions. This combination process is called crossover. Another form to obtain new solutions is through random changes in an existing chromosome called mutation. This process runs until a user-defined limit of generations is achieved.

Considering our particular problem, a chromosome (σ) is a complete tour containing all customers, and each customer is a gene. When a chromosome is generated, the Split algorithm, presented in Subsection 4.3.1, is employed to create feasible sub-tours (σ_j), and consequently, a feasible solution for IPDSP-P. The fitness value is given by the total distribution cost, as stated in equation (1) in Section 3. Figure 6 provides a example of a chromosome and its solution obtained from split algorithm.

The first-generation chromosomes are randomly generated. For the following generations, the roulette wheel selection process based on relative fitness selects a set of parents. Each couple of chromosomes generates two children by single-point crossover operator based on a crossover probability. Every child has a chance, given by mutation probability, to suffer mutation, performed by the SWAP algorithm. We used two stopping criteria:

1 a fixed number of generations

2 the number of iterations without improvement.

Figure 7 ilustrates the crossover and the mutation operators.

5 Computational study

5.1 Data generation

Random instances were generated by varying the number of customers and the normalised shelf life. Regarding the number of customers, both small and large instances were tested: 5, 7, 10, 15, 30 and 50 customers. The normalised shelf life was chosen based on the analysis presented in Subsection 5.2. Then, five random instances were generated for each combination of the number of customers and normalised shelf life, resulting in 90 distinct instances. The runtime was limited to simulate a realistic planning situation in which time limitation is present. The other parameters were randomly generated by the same procedure used in Amorim et al. (2013a). Tables 1 and 2 summarise the parameters considered for this study.

Table 1 Instances' parameters

Number of customers	5, 7, 10, 15, 30, 50
Execution time limits (s)	30, 60, 300 and 1,800 seconds
Normalised shelf life	short = 1.55
	long = 4.56
	very long = 10

Table 2 Procedure to generate the instances

Symbol	*Parameter*	*Generation method*
K	Number of vehicles	Equals to N (number of customers)
dem_c	Customer's demand	75% of demand follows a uniform distribution in the interval U[40, 60] and 25% is set to 0
pt_c	Production time (unit)	1
$CapV$	Vehicle capacity	$0.5 * \sum_c dem_c$
c, d	Node locations	Customers were positioned randomly in x-y plane from (0, 0) to (100, 100) and the plant is located at position (50, 50)
tt_{cd}	Travel times	Travel times were determined by the Euclidean distance between nodes
ct_{cd}	Variable costs	Variable costs are the same as the travel times
$\overline{ft}$	Fixed cost	250 for each vehicle used

5.2 Normalised shelf life definition

The shelf life of a perishable product may vary from hours to weeks. In fact, the shelf life in operational planning is relative and it is possible that a product that expires in two days can be considered less perishable than a product that expires in a week. For example, considering that the former product is delivered across the neighbourhood and the second is delivered all over the state, passing through several distribution centres, the product that expires in two days may be considered less perishable in an operational planning process.

Therefore, time is not sufficient to characterise the shelf life length in operational planning. Also, as we presented in Section 2, there are several different ways to formulate and define this parameter, which makes the distinction between a short and a long shelf life more confusing. As we did not found in the literature any common standard to characterise the shelf life across different researches, this study evaluated several forms to came up with a metric, the normalised shelf life, that could assist the characterisation of those (and further) scenarios containing perishable products.

To come up with that standard metric, we tested several candidates like the ratio between shelf life and the planning horizon, and the ratio between shelf life and the time of an average route, i.e., the average trip time, but we discarded both options because the variation of the planning horizon should not affect the perishability degree of the product and it should not be related to a decision variable of the problem.

Thus, we propose the following way to calculate the normalised shelf life:

$$Normalised\ shelf\ life = \frac{ShelfLife}{AvgTT} \tag{28}$$

where AvgTT is the average travel time from the depot (or plant) to customers. This is a good alternative to calculate the normalised shelf life because the travel time is a model's parameter and it shows how long a product can be stored in the most favourable case, which is when the product goes directly from depot to customer. Set AvgTT as the travel time between customers was also an option. However, that option was discarded because if all customers are far from the depot but close to each other, that could result in a false assumption of long normalised shelf life.

Equation (28) fits situations where the shelf life starts to count after the production finishes, however there are cases when the shelf life starts to count right after the production starts (Amorim et al., 2013a). In those cases, we also must consider the average processing time (AvgPT), as shown below:

$$Normalised\ shelf\ life = \frac{ShelfLife}{AvgTT + AvgPT} \tag{29}$$

To validate these metrics and to understand what is a short and what is a long shelf life, equations (28) and (29) were tested on several works in the literature, considering models containing either the vehicle routing or the direct shipment for the distribution component of the problem. The procedure to convert the shelf life to normalised shelf life for each paper is described in the following paragraphs. Table 3 summarises the obtained values after applying the equations.

Table 3 Normalised shelf life for several models in literature

Reference	*Short shelf life*	*Long shelf life*	*AvgTT*	*AvgPT*	*Short normalised shelf life*	*Long normalised shelf life*
Amorim et al. (2013a)	112.5	187.5	39	37.5	1.47	2.45
Amorim et al. (2012)	2	8	1	-	2	8
Li et al. (2016)	1	5	1	-	1	5
Li et al. (2020)	1	5	1	-	1	5
Marandi and Zegordi (2017)	96.92	241.53	69.80	-	1.39	3.46
Seyedhosseini and Ghoreyshi (2014)	2	3	1	-	2	3
Coelho and Laporte (2014)	2	5	1	-	2	5
Average without outliers					*1.55*	*4.56*

The models proposed by Amorim et al. (2012), Li et al. (2016, 2020), Seyedhosseini and Ghoreyshi (2014) and Coelho and Laporte (2014) considered a direct shipment delivery, and there was no delay between the decision and the delivery of the product. Thus, the AvgTT on those cases were set as 1.

To calculate the normalised shelf lives for (Amorim et al., 2013a; Marandi and Zegordi, 2017), we have generated a random instance following the instructions in the paper and calculated the shelf life, the AvgTT, and AvgPT based on the generated instance.

The short and long normalised shelf lives used in computational experiments (1.55 and 4.56, respectively) were obtained from the average of normalised shelf lives calculated from literature.

5.3 Parameter tuning for genetic algorithm

The selection of the genetic algorithm parameters may highly influence its performance. Because of that, the tuning of the parameters is necessary. In this study, we used the IRACE package (López-Ibáñez et al., 2016), which is an automatic procedure that selects the best set of parameters for a given experiment. The configuration of the IRACE was: the number of chromosomes, parents, and generations were selected from a range between 100 and 1,000. The crossover and mutation chances were selected from a range between 0 and 1. To feed the IRACE algorithm, a set of random instances were generated using the same procedure provided in Subsection 5.1. The tuned parameters provided by the algorithm are presented in Table 4.

Table 4 Parameters selected by IRACE package for genetic algorithm

Parameter	*Value*
Max population	790 chromosomes
Stopping criterion	87 generations
Crossover probability	73.86%
Mutation probability	01.27%

5.4 Computational results

5.4.1 Experiments setup

The MILP model and the LBBD were implemented with CPLEX 20.1 and were run on a PC with an Intel Core i5 8265U @ 1.6 GHz 1.8 GHz CPU and 8.0 GB of RAM. The genetic algorithm was run in the same PC, was implemented in Python, and each instance was run 30 times to assure statistical relevance of the results. The results are presented in two subsections. Subsection 5.4.2 is reserved for the smaller instances (5, 7, 10 and 15 customers), while Subsection 5.4.3 describes the results for the large instances (30 and 50 customers).

5.4.2 Small instances results

Since there were several instances for which both exact models could not find any feasible solution and the genetic algorithm found feasible solutions for every instance, we decided to present the results of the exact models separated from the genetic algorithm results. That visualisation provides a better understanding of how the shelf life impacts the expected performance of the exact solution methods.

For the exact approaches, i.e., the MILP and the LBBD models, we computed how many times the models could find the optimal solution, how many times they found any solution and how many times they found no solution. This result is presented in Table 5. As we are interested in the impact of the shelf life in those models, we grouped the data by the normalised shelf life and time limit. Thus, for each normalised shelf life and each time limit, 20 instances were solved (5 random instances * 4 different number of customers).

Table 5 MILP and LBBD results categorised by solution status, and grouped by normalised shelf life and time limit

Normalised shelf life	*Time limit (s)*	*MILP*			*Logic-based Benders decomposition*		
		No solution found	*Solution found*	*Optimal solution found*	*No solution found*	*Solution found*	*Optimal solution found*
1.55	30	10	5	5	10	5	5
	60	10	5	5	10	2	8
	300	10	5	5	6	4	10
	1,800	10	4	6	5	5	10
4.56	30	5	6	9		14	6
	60	3	7	10		13	7
	300	3	7	10		10	10
	1,800	3	7	10		10	10
10	30		10	10		15	5
	60		10	10		13	7
	300		10	10		10	10
	1,800		8	12		10	10

Table 6 Results for small instances using the genetic algorithm

Number of customers	*Normalised shelf life*	*GAP (%)*			
		30 s	*60 s*	*300 s*	*1,800 s*
5	1.55	0.00	0.00	0.00	0.00
	4.56	0.00	0.00	0.00	0.00
	10	0.00	0.00	0.00	0.00
7	1.55	29.01	29.01	29.01	29.01
	4.56	0.00	0.00	0.00	0.00
	10	0.00	0.00	0.00	0.00
10	1.55	87.51	87.51	87.51	87.51
	4.56	26.33	26.32	26.30	26.30
	10	21.62	21.62	21.61	21.61
15	1.55	78.50	78.50	78.50	78.50
	4.56	21.11	20.95	20.67	20.47
	10	20.65	20.46	20.26	20.06

Regarding the genetic algorithm, we computed the gap between the average solution from all runs and the instance lower bound provided by CPLEX after running the solver for 3,600 s. Equation (30) shows how the GAP was calculated, and Table 6 summarises the results.

$$GAP = \left| \frac{Avg(GA\_runs)}{CPLEX\_Lower\_Bound} - 1 \right| \tag{30}$$

Figure 8 Boxplot of random instance '1' with 15 customers for genetic algorithm computational experiment (see online version for colours)

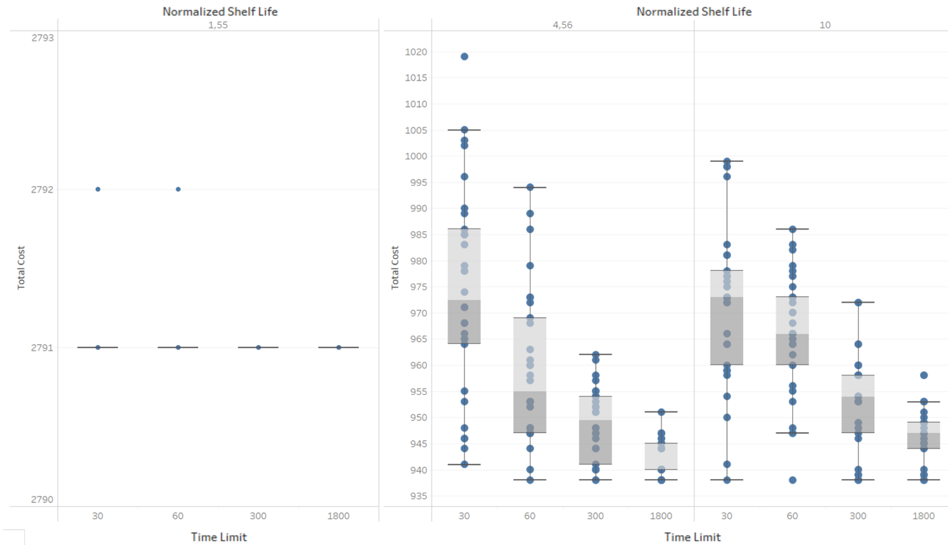
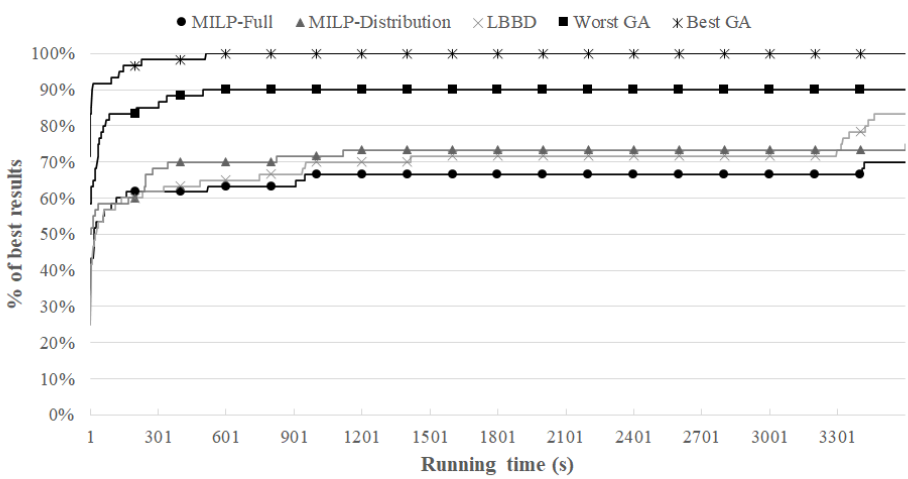


Figure 9 Relationship between runtime and the number of instances for which the best result was found



One impact of shelf life on the genetic algorithms is that they converge faster to a single solution when the shelf life is shorter. This behaviour is observed when we plot the data from the thirty runs of the same instance in a boxplot. Boxplot is a visualisation tool that, given a dataset, allows us to identify the middle and how spread out from the middle the data is (Frigge et al., 1989). Figure 8 shows a boxplot for one of the instances with 15 customers. In that example, for the short NSL (1.55), it is observed that almost all runs converge to a single value regardless of the time limit. For longer shelf lives (NSL = 4.56 and 10), we observed as the time limit increased, the median value and the dispersion across runs improved. Similar behaviour can be found for the other instances.

Finally, we provide a performance comparison among all models developed in this study. To perform this test, we evaluated all previous tests to get the best result obtained among the three solution methods for each of the 60 instances. Then, we run the models one more time and registered the runtime each model took to achieve the best result. As the genetic algorithm has a stochastic component, we had run it 30 times for each instance and computed the best and the worst runtime. Also, the runtime was limited to 3,600 s for all models. The results of the performance test are presented in Figure 9.

5.4.3 Large instances results

Regarding the performance of the exact methods for 30- and 50-customer instances, the MILP could not find a feasible result for any instance at all, while the LBBD only found feasible results for 30 customers' instances, however, the optimality could not be proven for any of them. Table 7 presents the results for the LBBD on large instances.

Table 7 LBBD results for 30 customers' instances categorised by solution status, and grouped by normalised shelf life and time limit

Normalised shelf life	*Time limit (s)*	*No solution found*	*Solution found*	*Optimal solution found*
1.55	30	5	0	0
	60	5	0	0
	300	5	0	0
	1,800	5	0	0
4.56	30	4	1	0
	60	4	1	0
	300	4	1	0
	1,800	3	2	0
10	30	4	1	0
	60	4	1	0
	300	2	3	0
	1,800	1	4	0

Table 8 Results for large instances using the genetic algorithm

Number of customers	*Normalised shelf life*	*GAP (%)*			
		30 s	*60 s*	*300 s*	*1,800 s*
30	1.55	78.71	78.32	77.85	77.45
	4.56	20.11	19.42	18.94	18.40
	10	14.46	13.69	12.99	12.43
50	1.55	80.62	79.85	78.97	78.35
	4.56	23.41	22.74	22.08	21.54
	10	13.32	12.97	12.45	11.90

Regarding the genetic algorithm, we also computed the gap as presented in equation (30), whose results are summarised on Table 8.

Figure 10 Boxplot of random instance '1' with 30 customers for genetic algorithm computational experiment (see online version for colours)

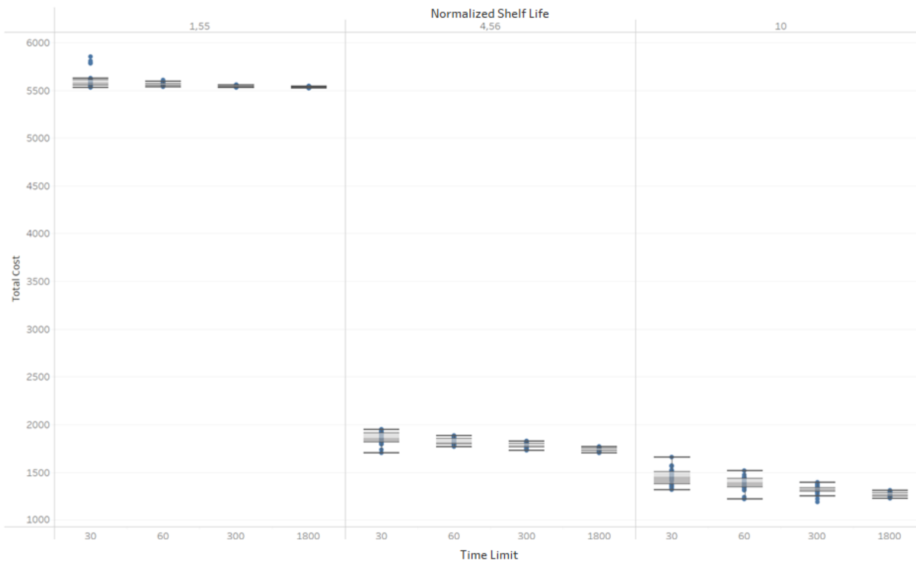
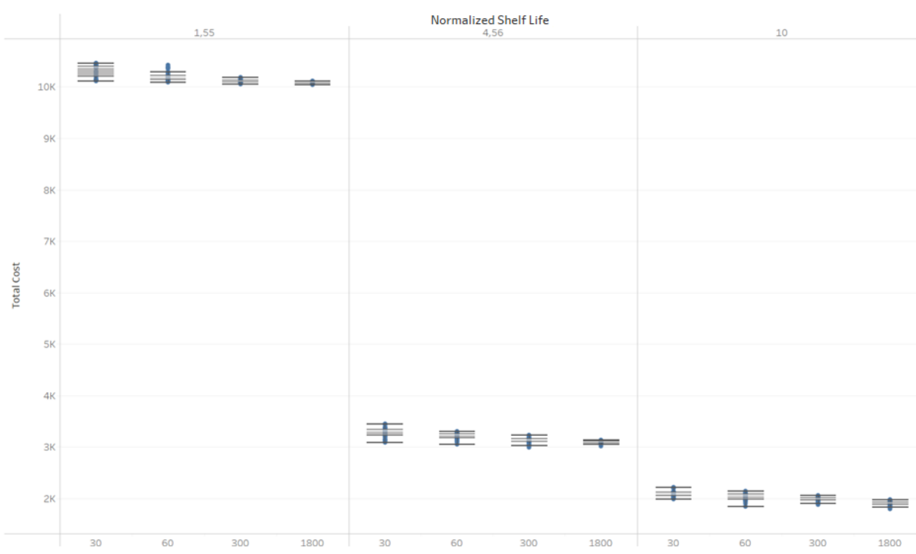


Figure 11 Boxplot of random instance '1' with 50 customers for genetic algorithm computational experiment (see online version for colours)



We also plotted the boxplot to support the understanding of the influence of shelf life and the time limit on the median and the dispersion of results across all runs from a same large instance. Figures 10 and 11 show, respectively, an example of an instance with 30 and 50 customers. When compared to the small instances, the major difference seem on the genetic algorithm results is that it did not converged to a single value when the NSL = 1.55. Instead, it has behaviour similar to the other NSL levels, i.e., there was a improvement both on the median and on the dispersion. This behaviour could be observed for all large instances experiments.

5.5 Discussion

Although several studies have shown that the integrated approach leverages the results for products with shorter shelf lives (Amorim et al., 2012; Farahani et al., 2012), Tables 5 and 7 also show that a shorter shelf life makes the IPDSP-P harder to solve using exact approaches. This conclusion is based on the increase in the number of instances for which no solution was found as the shelf life gets shorter, which happens for both the MILP and LBBD models.

The same conclusion cannot be drawn for the genetic algorithm. Although the GAPs get higher as the shelf life gets shorter for both small and large instances (Tables 6 and 8), we also noticed that the algorithm converged faster when the shelf life was shorter and the instance has a small size (Figure 8). This can indicate one of two things:

1 As can be seen in Table 2, the number of vehicles is not a constraint of the problem. Therefore, the split algorithm will always find a feasible solution. However, the solutions may be converging to a local optimum, which explains why the GAP is higher for the shorter shelf lives and small instances.

2 Since the shorter shelf life makes it harder to find a feasible solution by means of an exact approach, the lower bound found in CPLEX may be very unrealistic for the short life situation.

The explanation for the difficulty to find a feasible solution using exact models when the shelf life is very short may be due to the lower number of feasible solutions, and any small change in the solution of a relaxed problem can violate the shelf life constraint, which turns the problem infeasible. This may be the same reason why the genetic converges so fast to a single solution. Therefore, we may conclude that solution methods that have good mechanisms to avoid the solutions to get stuck in local optima are good choices when handling the IPDSP-P with a short shelf life product.

Therefore, the shelf life of a product is an important variable to be taken into account when deciding between an integrated approach, or a decoupled one, or even an iterative procedure, like what was done in Farahani et al. (2012) and Seyedhosseini and Ghoreyshi (2014).

When we compare the exact models and the genetic algorithm, considering both short and long runtime, the genetic algorithm had found the best results than the other models, as seen in Figure 9. This result may be explained by the difficulty for the exact models to find even a feasible solution for instances where the shelf life is short and for bigger instances. On the other hand, the split algorithm always provides a feasible solution for the problem studied in this paper, which benefits the genetic algorithm.

Comparing the exact methods: for short running times, the MILP model results were slightly better than the LBBD approach. However, as the running time increases the LBBD model had a better improvement in the results. The functioning of the LBBD approach may explain that. The LBBD breaks the problem into a master problem and subproblems, which may cause many infeasible solutions to be considered at the beginning of the execution. However, as the model runs for more time, the subproblems eliminate many infeasible solutions, and the master problem's solution space becomes more resembling to IPDSP-P's solution space. However, the master problem still is a capacitated vehicle routing problem (CVRP), which is already a difficult problem to solve. That may explain why the genetic algorithm outperforms the LBBD model even when we let the models running for a long time.

6 Conclusions and future research

In this paper, we investigated the relationship between the shelf life length and the difficulty of solving the integrated production-distribution scheduling problem (IPDSP-P). We considered a single machine environment that produces one perishable product that must be delivered by homogenous vehicles. There are two decisions regarding the routing strategy: the allocation of client orders to vehicles and the delivery sequence of each route. This paper proposes and implements a MILP model, a logic-based Benders decomposition model, and a genetic algorithm to solve this problem. Several computational experiments were conducted by varying the number of customers, the normalised shelf life, and the time available to run the model.

The normalised shelf life is a metric we proposed to allow the conversion of the various forms to calculate the shelf life found in literature into a single comparable measure. Therefore, this metric may be valuable in future studies about operational planning for perishable products.

Our experiments showed that when the normalised shelf life is short, the number of instances for which both MILP and LBBD models could not find a feasible solution was higher when compared to the instances with long or very long normalised shelf life. That result contributes to the conclusion that the shelf life influences the difficulty of solving the IPDSP-P by exact models. Regarding the genetic algorithm, our experiments showed that the GAP between the lower bound obtained for the objective function and the genetic algorithm solution was higher for the short shelf life instances. However, we also found that the solutions converge to a single solution faster, in those cases. This result may indicate that the genetic algorithm got stuck in a local optimum.

There are still plenty of opportunities for future research. For example, one hypothesis for why the short shelf life makes the IPDSP-P harder to solve is that even minor adjustments in a solution may violate the shelf life constraint, which turns the problem infeasible. That may lead to the conclusion that the feasible solutions are more scattered over the solution space. Therefore, combining linear programming to heuristics (or metaheuristics) that turns the solution search process broader over the solution space could be a good research direction. That model could allow one to find the optimal solutions for bigger instances of the problem. Another research direction is the analysis of the influence of shelf life on more complex scenarios (or even empirical situations).

Acknowledgements

This work was supported by Conselho Nacional de Desenvolvimento Científico e Tecnológico under grants 405702/2021-3 and 310812/2021-6; FAPESP – Fundacao de Amparo a Pesquisa do Estado de Sao Paulo under grants 2016/01860-1 and 2019/12023-1.

References

Amorim, P., Belo-Filho, M., Toledo, F., Almeder, C. and Almada-Lobo, B. (2013a) 'Lot sizing versus batching in the production and distribution planning of perishable goods', *International Journal of Production Economics*, Vol. 146, No. 1, pp.208–218.

Amorim, P., Gunther, H-O. and Almada-Lobo, B. (2012) 'Multi-objective integrated production and distribution planning of perishable products', *International Journal of Production Economics*, Vol. 138, No. 1, pp.89–101.

Amorim, P., Meyr, H., Almeder, C. and Almada-Lobo, B. (2013b) 'Managing perishability in production-distribution planning: a discussion and review', *Flexible Services and Manufacturing Journal*, Vol. 25, No. 3, pp.389–413.

Chandra, P. and Fisher, M.L. (1994) 'Coordination of production and distribution planning', *European Journal of Operational Research*, Vol. 72, No. 3, pp.503–517.

Chen, Z-L. (2010) 'Integrated production and outbound distribution scheduling: review and extensions', *Operations Research*, Vol. 58, No. 1, pp.130–148.

Chu, Y. and Xia, Q. (2004) 'Generating Benders cuts for a general class of integer programming problems', *International Conference on Integration of Artificial Intelligence (AI) and Operations Research (OR) Techniques in Constraint Programming*, Springer, pp.127–141.

Coelho, L.C. and Laporte, G. (2014) 'Optimal joint replenishment, delivery and inventory management policies for perishable products', *Computers and Operations Research*, Vol. 47, pp.42–52.

Dantas, H.T.A. (2022) *The Influence of Shelf Life on the Integrated Production Scheduling and Vehicle Routing Optimization for Perishable Products*, Master's thesis, Universidade Federal de São Carlos.

Dantzig, G.B. and Ramser, J.H. (1959) 'The truck dispatching problem', *Management Science*, Vol. 6, No. 1, pp.80–91.

Devapriya, P., Ferrell, W. and Geismar, N. (2017) 'Integrated production and distribution scheduling with a perishable product', *European Journal of Operational Research*, Vol. 259, No. 3, pp.906–916.

Di Placido, A., Archetti, C. and Cerrone, C. (2022) 'A genetic algorithm for the close-enough traveling salesman problem with application to solar panels diagnostic reconnaissance', *Computers & Operations Research*, Vol. 145, p.105831.

Farahani, P., Grunow, M. and Gunther, H-O. (2012) 'Integrated production and distribution planning for perishable food products', *Flexible Services and Manufacturing Journal*, Vol. 24, No. 1, pp.28–51.

Fikar, C., Mild, A. and Waitz, M. (2021) 'Facilitating consumer preferences and product shelf life data in the design of e-grocery deliveries', *European Journal of Operational Research*, Vol. 294, No. 3, pp.976–986.

Frigge, M., Hoaglin, D.C. and Iglewicz, B. (1989) 'Some implementations of the boxplot', *The American Statistician*, Vol. 43, No. 1, pp.50–54.

Garcia, D.J. and You, F. (2015) 'Supply chain design and optimization: challenges and opportunities', *Computers & Chemical Engineering*, Vol. 81, pp.153–170.

Grossmann, I. (2005) 'Enterprise-wide optimization: a new frontier in process systems engineering', *AIChE Journal*, Vol. 51, No. 7, pp.1846–1857.

Grossmann, I.E. (2012) 'Advances in mathematical programming models for enterprise-wide optimization', *Computers & Chemical Engineering*, Vol. 47, pp.2–18.

Hooker, J.N. and Ottosson, G. (2003) 'Logic-based Benders decomposition', *Mathematical Programming*, Vol. 96, No. 1, pp.33–60.

Li, J., Gao, X., Guo, B. and Wu, M. (2020) 'Production plan for perishable agricultural products with two types of harvesting', *Information Processing in Agriculture*, Vol. 7, No. 1, pp.83–92.

Li, Y., Chu, F., Yang, Z. and Calvo, R.W. (2016) 'A production inventory routing planning for perishable food with quality consideration', *IFAC-PapersOnLine*, Vol. 49, No. 3, pp.407–412.

López-Ibáñez, M., Dubois-Lacoste, J., Pérez Cáceres, L., Stützle, T. and Birattari, M. (2016) 'The irace package: iterated racing for automatic algorithm configuration', *Operations Research Perspectives*, Vol. 3, pp.43–58.

Marandi, F. and Zegordi, S.H. (2017) 'Integrated production and distribution scheduling for perishable products', *Scientia Iranica*, Vol. 24, No. 4, pp.2105–2118.

Moons, S., Ramaekers, K., Caris, A. and Arda, Y. (2017) 'Integrating production scheduling and vehicle routing decisions at the operational decision level: a review and discussion', *Computers and Industrial Engineering*, Vol. 104, pp.224–245.

Mula, J., Peidro, D., Diaz-Madronero, M. and Vicens, E. (2010) 'Mathematical programming models for supply chain production and transport planning', *European Journal of Operational Research*, Vol. 204, No. 3, pp.377–390.

Papageorgiou, L.G. (2009) 'Supply chain optimisation for the process industries: advances and opportunities', *Computers & Chemical Engineering*, Vol. 33, No. 12, pp.1931–1938.

Prins, C. (2004) 'A simple and effective evolutionary algorithm for the vehicle routing problem', *Computers and Operations Research*, Vol. 31, No. 12, pp.1985–2002.

Reimann, M., Tavares Neto, R. and Bogendorfer, E. (2014) 'Joint optimization of production planning and vehicle routing problems: a review of existing strategies', *Pesquisa Operacional*, Vol. 34, No. 2, pp.189–214.

Seyedhosseini, S. and Ghoreyshi, S. (2014) 'An integrated model for production and distribution planning of perishable products with inventory and routing considerations', *Mathematical Problems in Engineering*, Vol. 2014.

Shin, H., Lee, T. and Lee, H-R. (2022) 'Skyport location problem for urban air mobility system', *Computers & Operations Research*, Vol. 138, p.105611.

Solina, V. and Mirabelli, G. (2021) 'Integrated production-distribution scheduling with energy considerations for efficient food supply chains', *Procedia Computer Science*, Vol. 180, pp.797–806.

Türkyılmaz, A., Senvar, O., Ünal, İ. and Bulkan, S. (2022) 'A hybrid genetic algorithm based on a two-level hypervolume contribution measure selection strategy for bi-objective flexible job shop problem', *Computers & Operations Research*, Vol. 141, p.105694.

Vidal, C.J. and Goetschalckx, M. (1997) 'Strategic production-distribution models: a critical review with emphasis on global supply chain models', *European Journal of Operational Research*, Vol. 98, No. 1, pp.1–18.