



HAL
open science

Production routing decisions in a two-echelon supply chain with multiple delivery modes

Rachida Benfedel, Fayçal Belkaid, Nadjib Brahim

► To cite this version:

Rachida Benfedel, Fayçal Belkaid, Nadjib Brahim. Production routing decisions in a two-echelon supply chain with multiple delivery modes. *International Transactions in Operational Research*, 2025, 33, pp.3385 - 3421. <10.1111/itor.70019>. <hal-05597777>

HAL Id: hal-05597777

<https://hal.science/hal-05597777v1>

Submitted on 21 Apr 2026

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons CC BY 4.0 - Attribution - International License

WILEY

INTERNATIONAL
TRANSACTIONS
IN OPERATIONAL
RESEARCHIntl. Trans. in Op. Res. 33 (2026) 3385–3421
DOI: 10.1111/itor.70019

Production routing decisions in a two-echelon supply chain with multiple delivery modes

Rachida Benfedel^{a,b,*} , Fayçal Belkaid^a and Nadjib Brahim^b^a *Manufacturing Engineering Laboratory of Tlemcen, University of Tlemcen, 22, Rue Abi Ayed Abdelkrim Fg Pasteur B.P 119, Tlemcen 13000, Algeria*^b *Rennes School of Business, 2 rue Robert d'Arbrisel, Rennes 35065, France**E-mail: rachida.benfedel@rennes-sb.com [Benfedel]; faycal.belkaid@univ-tlemcen.dz [Belkaid]; nadjib.brahimi@rennes-sb.com [Brahimi]*

Received 7 March 2024; received in revised form 7 January 2025; accepted 20 February 2025

Abstract

We study an original two-echelon production routing problem with multiple delivery modes (2E-PRP-MDM). In the first echelon, the primary production facility is tasked with satisfying the demands of two distinct entities: a set of warehouses and a set of customers through direct shipments. In the second echelon, warehouses become delivery hubs, meeting the needs of another group of customers with the possibility of delivering several customers using the same vehicle, which entails routing decisions. The use of two different delivery modes is justified by the segmentation of customers into very important customers with high volumes (first echelon) and low-volume customers (second echelon). This supply chain structure requires efficient decisions on production planning, routing, and practical resource allocation across both echelons. In order to address this challenging optimization problem, we have developed a mixed-integer linear programming (MILP) formulation and two heuristics, a three-phase iterative (3PI) approach, and a simulated annealing (SA) algorithm with path relinking. Extensive numerical experiments showed that the simulated annealing heuristic with path relinking (PR) outperforms the 3PI approach, which in turn outperforms the branch-and-cut procedure of the solver when run on the MILP formulation. In terms of managerial insights, we compared scenarios with and without delivery flexibility from the manufacturing to warehouses to the customers. Our findings provide valuable insights for managers seeking to reduce costs, based on the analysis of the supply chain of an Algerian soft drink company. The results show that an integrated approach achieves a cost reduction of over 2.73% compared to the current practice of the company. Additionally, they reveal that increasing the number of warehouses does not always lead to lower total costs.

Keywords: production planning; local search; mixed-integer linear programming; two-echelon production routing problem

*Corresponding author.

© 2025 The Author(s).

International Transactions in Operational Research published by John Wiley & Sons Ltd on behalf of International Federation of Operational Research Societies.

This is an open access article under the terms of the Creative Commons Attribution-NonCommercial-NoDerivs License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.

1. Introduction

Coordination and integration of decisions in supply chain planning are key strategies to enhance supply chain profitability (Khan et al., 2014). Even before the emergence of supply chain management concepts, it was emphasized by Bowersox (1969) that decision integration should be between different stages of the product flow. Which can be translated in nowadays terms by the concept of vertical supply chain integration. Such integration is motivated by increasing efficiency and by the booming of e-commerce and smart retail stores, and it is supported by new business practices such as the vendor-managed inventory and technological enablers such as the industrial Internet of Things. A recent survey by Hrabec et al. (2022) showed that decision integration can result in cost savings of up to 28.03% and 11.1% on average. Nevertheless, decision integration results in a variety of challenging optimization problems.

However, full integration of decisions throughout all supply chain stages can be extremely challenging from both a practical and managerial point of view as well as from an optimization perspective. The decision integration requires coordination between decision-makers in different supply chain stages. Such coordination is challenging when the decision-makers belong to different departments in the same company, much less when they belong to different organizations. From an optimization point of view, considering all decisions in a single monolithic model results in a very complex optimization problem that can be extremely difficult to solve.

Hence, in the OR/MS literature, there exist several studies inspired by real industrial applications that consider the integration of decisions between a few processes or departments. Examples include lot sizing with transportation and facility location (e.g., Ambrosino and Scutellá, 2005), production planning with quality inspection (e.g., Khan et al., 2014), and maintenance and operations scheduling (e.g., Pandey et al., 2011). One of the most challenging problems with interesting applications is the production routing problem (Adulyasak et al., 2015b).

This study was inspired and driven by the challenges faced by a manufacturer of personal protective equipment. During the COVID-19 pandemic, the company experienced a significant surge in demand, exposing substantial limitations in its supply chain. These include a decoupled approach to production and distribution planning, a retailer-managed inventory system that results in high costs and insufficient transportation capacity, and a reliance on manual planning methods.

To address these challenges, we developed and solved an extension of the production routing problem, namely a two-echelon production routing problem with mixed delivery modes (2E-PRP). This work was motivated by the lack of studies specifically addressing two-echelon PRP, which are essential for optimizing complex supply chains. The 2E-PRP is a complex optimization problem that seeks to minimize the total cost of production, inventory, and transportation in a two-echelon supply chain. The first echelon comprises the production plants, and the second echelon consists of the distribution centers. The objective is to determine the production quantities at each facility, the inventory levels at each distribution center, and the routing of vehicles between the facilities and distribution centers. The use of two different delivery modes is justified by the fact that companies split their customers into very important customers with high volumes and low volume customers. In the latter category, several customers can be delivered using the same vehicle. The former and most important category of customers is delivered through direct shipment mode. This novel 2E-PRP is more challenging than the production routing problem, but it responds in a better way to the practical needs of solving such integrated optimization problems for many companies.

This paper aims to study a 2E-PRP with multiple delivery modes. Specifically, our study seeks to (1) Formulate an original and challenging 2E-PRP as a mixed-integer linear program (MILP), (2) adapt an existing heuristic from the literature as a first approach to solve the problem, (3) develop a new efficient heuristic that outperforms both the adapted heuristic and a commercial MILP solver, and (4) propose several managerial insights.

This paper's structure is as follows. Section 2 offers an overview of the relevant literature on the subject. Section 3 describes the problem under investigation and the mathematical model used to address it. Section 4 details the expanded two-iterative phase heuristic and the simulated annealing with path relinking (SA-PR). Section 5 presents the results of numerical experiments and computational analysis, providing insights for managers and examining the algorithm's performance. Conclusions and discussions of future research are presented in Section 6.

2. Literature review

This literature review section groups related studies into four parts: background on the production routing problem (PRP), single-echelon PRP, two-echelon PRP (2E-PRP), and the most used solution approaches. This will allow us to position our work, highlight our contributions, and justify our choices in terms of a solution approach.

The roots of the 2E-PRP can be traced back to the broader domain of PRPs. The seminal work by Wagner and Whitin (1958) on the lot-sizing problem (LSP) and Dantzig and Ramser (1959) on the vehicle routing problem (VRP) laid the foundation for understanding and optimizing production and routing problems. Thousands of articles have been published on the VRP and the LSP. The VRP was reviewed by Laporte (2009) and Mor and Speranza (2022) among others. The most recent survey on lot sizing problems is available in the work of Brahimi et al. (2017) with a focus on single-item models.

2.1. The single-echelon and the two-echelon PRP

Before delving into the specifics of the two-echelon problem, it is essential to review studies addressing single-echelon PRPs. Noteworthy contributions include Chandra (1993), who focused on minimizing setup, production, inventory, and transportation costs. Subsequent research by Chandra and Fisher (1994) highlighted the substantial benefits of integrating production and routing decisions, emphasizing cost savings compared to separate optimization of LSP and VRP components. Recently, Hrabec et al. (2022) showed that integration of production planning and routing decisions in the PRP results in cost savings of 11.1% on average. There has been a rapid increase in the number of publications on the PRP in the last two decades. The publications present extensions and different variants of the problem, real applications, or improved solution approaches. Example extensions and variants are single-item (e.g., Archetti et al., 2011), multi-item (e.g., Fumero and Vercellis, 1999), backordering (e.g. Brahimi and Aouam, 2016), and remanufacturing (e.g. Qiu et al., 2018). Studies on the PRP have been seen in different industries, including fish feed industry (Brekka et al., 2022), meat production (Neves-Moreira et al., 2019), petrochemical industry (Schenekemberg et al., 2021), and home delivery services (Qiu et al., 2021). Ramos et al. (2022) introduced an arc flow model to address the multi-trip production, inventory, distribution, and

routing problem with time windows. Solution approaches will be discussed in more detail later in this section. For a comprehensive survey of publications on the PRP, see Adulyasak et al. (2015b) and Hrabec et al. (2022).

A particularly interesting extension of the PRP for our study is PRP with multiple plants. Lei et al. (2006) presented a comprehensive model for such a problem while considering a heterogeneous fleet of vehicles. The plants are further constrained with limited production and storage capacities. The authors addressed this complex PRP problem by introducing a two-phase decomposition heuristic to optimize direct shipments and consolidation inefficiencies. The second study on multiple plant PRP was done by Li et al. (2020). The authors considered the PRP within the framework of perishable food products. Fix-and-optimize and branch-and-cut algorithms were used to solve the problem. To the best of our knowledge, no other studies besides Lei et al. (2006) and Li et al. (2020) have addressed models with multiple plants.

The 2E-PRP was studied by Schenekemberg et al. (2021) and Qiu et al. (2021). These are, to the best of our knowledge, the only publications on this topic. Schenekemberg et al. (2021) developed a 2E-PRP model for a vendor-managed inventory system in the petrochemical industry, integrating pickups, production, inventory, and delivery decisions to minimize costs related to routing, production, and inventory. Their model involves vehicles transporting raw materials from suppliers to plants and finished products from plants to customers, with each vehicle performing at most one pickup and/or one delivery per period and all routes starting and ending at the plant. Schenekemberg et al. (2021) address both maximum level and order-up-to policies for deliveries. Qiu et al. (2021) examined the 2E-PRP with cross-docking satellites pertinent to e-commerce and city logistics. Production plants route products to cross-docking satellites, which then deliver to customers without storage capabilities. Our study, however, focuses on the two-echelon production routing problem with multiple delivery modes (2E-PRP-MDM). In our model, customers are divided into primary and secondary sets. In the first echelon, manufacturing sends a direct shipment to primary customers and warehouses, whereas in the second echelon, the warehouse routes secondary customers.

2.2. Solution approaches

Solution approaches suggested for the PRP fall into either exact methods or heuristic approaches that can be classified as decomposition heuristics, mixed-integer programming (MIP) based heuristics, or metaheuristics. Despite the very high complexity of production routing problems, several authors have proposed to solve them using exact methods. The main purpose of such methods is to analyze the performance of the other developed heuristics and metaheuristics by the same authors. All known exact methods developed for the PRP are based on a branch-and-cut procedure. These include the papers by Ruokokoski et al. (2010), Archetti et al. (2011), and Adulyasak et al. (2014a). These procedures take several hours to discover optimal solutions. For example, the algorithm of Adulyasak et al. (2014a) finds the optimal solution for instances with 35 customers, three time periods, and three vehicles in two hours.

Decomposition heuristics were proposed by Chandra (1993), Chandra and Fisher (1994), Boudia et al. (2008), Neves-Moreira et al. (2019), and Chekoubi et al. (2022). The principal idea of these heuristics is to decompose the problem into production and routing subproblems. The subproblems are usually solved sequentially before calling a heuristic that enhances the solution. The originality

of the different heuristics comes from the approach used to improve the resulting solution and the order in which the subproblems are solved. Mathematical programming-based heuristics were used to solve the PRP by Absi et al. (2015) and Brahimi and Aouam (2016), among others. The approach of Brahimi and Aouam (2016) is a relax-and-fix heuristic mixed with a local search procedure. For Brahimi and Aouam (2016), in each iteration of the relax-and-fix procedure, lot-sizing decisions are taken before routing decisions. Absi et al. (2015) proposed an iterative mathematical programming heuristic where routing costs are replaced in the formulation of the original PRP model with fixed costs approximating the cost of visiting a customer by a vehicle in a given time period. At each iteration, the resulting MIP is solved to optimality, the routing decisions are obtained using a heuristic, and the approximate visiting costs are updated before starting the iteration. Among the metaheuristics used to solve the PRP, we can cite the greedy randomized adaptive search procedure or GRASP (Boudia et al., 2007), the memetic algorithm (Boudia and Prins, 2009), tabu search (Bard and Nananukul, 2009; Shiguemoto and Armentano, 2010), and the adaptive large neighborhood search heuristic or ALNS (Adulyasak et al., 2014b). Both papers on the 2E-PRP (Schnekemberg et al., 2021, Qiu et al., 2021) suggest exact approaches based on the branch-and-cut algorithm to solve the problem. However, Schnekemberg et al. (2021) also suggest a novel exact algorithm that uses parallel computing and local search, while Qiu et al. (2021) propose a matheuristic to provide an initial feasible solution. The main interest in using such exact algorithms is their usefulness in analyzing the quality of heuristic solutions in addition to the fact that they allow to drawing of more reliable managerial insight based on optimal solutions. However, it should be noted that these algorithms might take hours to reach optimality even with relatively small-size instances. This is not acceptable for practical applications.

A discussion of the different solution approaches and their performance can be found in Adulyasak et al. (2015b). They have concluded that metaheuristics can be an excellent compromise between solution quality and CPU time in particular the ALNS proposed by the same authors. However, the approach proposed by Absi et al. (2015) outperforms the ALNS in terms of solution quality. SA has been successfully applied to a variety of hard combinatorial optimization problems, such as the lot sizing problem (Ceschia et al., 2017), but its application to solving production routing problems is limited. The combined SA-PR has never been applied to the PRP. As a result, we applied the SA-PR to the two-echelon production routing problem. As the numerical experiments show, this approach is very efficient for such problems.

In this study, we propose to solve the problem using three different approaches. The first approach consists of running a commercial solver on the mixed integer linear programming formulation (MILP) presented in Section 3. Second, we present a three-phase heuristic by adapting and extending the two-phase heuristic proposed by Absi et al. (2015). Finally, the third approach consists of a SA procedure with path relinking (SA-PR). Both heuristics present a good compromise between CPU time and solution quality. However, the SA-PR procedure outperforms the three-phase heuristic.

3. Problem description and formulation

We consider a single-item production routing problem. The distribution network is represented with an undirected graph $G = (N, A)$, where N is the union of four distinct sets: N_0 , N_c , N_w , and $N_{c'}$.

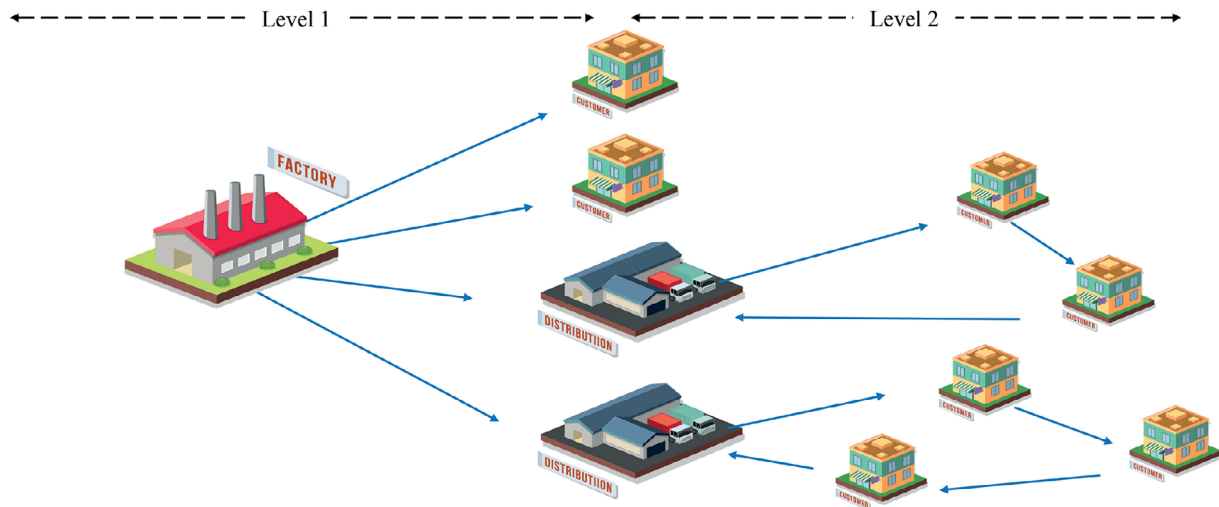


Fig. 1. Graphical illustration of supply network.

Set N_0 represents the nodes corresponding to the manufacturing plants, N_c represents the set of customers served by the plant, N_w denotes the set of warehouses, and $N_{c'}$ corresponds to the set of customers served by the warehouses. The edges in the graph are denoted as A . The network is illustrated in Fig. 1. The problem involves two echelons. In the first echelon, composed of nodes $N_0 \cup N_c \cup N_w$, the manufacturing plant serves the customers in N_c and the warehouses in N_w through direct shipments. An unlimited number of vehicles facilitates this, each with a capacity of Q_0 , and the transportation cost from the plant to $i \in N_c \cup N_w$, is represented as c_i . In the second echelon (nodes $N_w \cup N_{c'}$), the warehouses take responsibility for serving the customers in $N_{c'}$ using a fleet of a limited number of homogeneous vehicles, denoted as K . Each vehicle has a capacity of Q . As indicated in previous studies (e.g., Adulyasak et al., 2014a), each customer is delivered once by a given vehicle in a given period. Also, similarly to Qiu et al. (2021), we assume that a customer can be served by only one warehouse in a given time period. Notice that the warehouse plays an important role in aggregating demand across locations and times. It is used to benefit from economies of scale and help reduce setup costs. In some cases, such a warehouse can have lower per-unit inventory holding costs (due to size and specialization); hence, it can be an alternative to storing products at the company itself. Also, due to limitations of storage capacity at the plant, the warehouse should be used to store extra production. The transportation cost from warehouse i to customer j is represented as c_{ij} , where $i, j \in N_w \cup N_{c'}$. The planning horizon is limited to $|T|$ time periods. The manufacturing plant operates with a capacity of C in each period $t \in T$, incurring a production cost p and a setup cost f . Each node $i \in N$ is assigned a unit inventory cost h_i , an initial inventory level I_{i0} , and a maximum inventory level U_i . In every period $t \in T$, customer $i \in N_c \cup N_{c'}$ orders d_{it} units of the product.

The decisions to be taken, for each time period, are to determine

- the levels of production at the manufacturing units,
- the quantities being transported by the vehicles from the manufacturing units to the warehouses and customers in set N_c and from the warehouses to customers in set $N_{c'}$,

- the order in which customers of the second echelon are visited (N_c),
- and consequently the levels of inventory at the end of each time period for the manufacturing units, the warehouses, and the customers.

The objective is to minimize the total production and logistics costs. In this paper, we assume there is only one single manufacturing plant. An extension of the model to multiple plants is straightforward. We introduce the following notation:

Sets and parameters:

- Sets:
 - $T = \{1, \dots, |T|\}$: set of time periods in the planning horizon.
 - $N = \{N_0 \cup N_c \cup N_w \cup N_c'\}$: set of nodes (processing unit, customers in the first echelon, warehouses, and customers in the second echelon) where $N_0 = \{0\}$, $N_c = \{1, \dots, |N_c|\}$, $N_w = \{|N_c| + 1, \dots, |N_c| + |N_w|\}$, and $N_c' = \{|N_c| + |N_w| + 1, \dots, |N_c| + |N_w| + |N_c'|\}$.
 - $A = \{A_1 \cup A_2\}$: set of arcs, where $A_1 = \{(0, i) : i \in N_c \cup N_w\}$ and $A_2 = \{(i, j) : i, j \in N_c', i \neq j \vee i \in N_c', j \in N_w \vee i \in N_w, j \in N_c'\}$.
 - $K = \{1, \dots, |K|\}$: set of homogeneous vehicles at the warehouses.
- Production data:
 - p : Production cost.
 - f : Setup cost.
 - C : Production capacity.
- Inventory data:
 - h_i : Inventory holding cost at node $i \in N$.
 - U_i : Storage capacity at location $i \in N$.
 - I_{0i} : Initial inventory at location $i \in N$.
 - d_{it} : Demand of customer $i \in \{N_c \cup N_c'\}$ in period t .
- Transport data:
 - c_i : Travel cost from manufacturing plant to warehouses and customer $i \in N_w \cup N_c$.
 - c_{ij} : Travel cost between $(i, j) \{i, j \in N_w \cup N_c'\}$.
 - Q_0 : manufacturing plant's vehicle capacity.
 - Q_w : warehouses vehicle capacity.

Decision variables:

- Production:
 - x_t : Production level at period t .
 - y_t : Binary setup variable equal to 1 if there is a production in period t and 0 otherwise.
- Inventory:
 - I_{it} : Inventory level of the product at location i in period t .
- Transport:
 - v_{it} : Number of vehicles sent to node $i \in N_c \cup N_w$ in period t .
 - q_{it} : Amount of product shipped to node $i \in N_c \cup N_w$ in period t
 - r_{ijkt} : Amount delivered from warehouse i to customer $j \in N_c'$ by vehicle k at period t
 - L_{wijkkt} : amount of load carried by vehicle k of warehouse w when arc $(i, j) \in A_2$ is traversed in period t .

H_{ijkt} : binary variable, equal to 1 if vehicle k serves customer $j \in N_c$ from warehouse i in period t , 0 otherwise.

Y_{wijkt} : binary variable, equal to 1 if arc $(i, j) \in A_2$ is traversed in period t by vehicle k of warehouse w , 0 otherwise.

The proposed MILP formulation is presented below.

$$\begin{aligned}
 [2E - PRP - MDM] \min \quad & \sum_{t \in T} px_t + \sum_{t \in T} fy_t + \sum_{i \in N} \sum_{t \in T} h_i I_{it} + \sum_{t \in T} \sum_{i \in N_c \cup N_w} c_i v_{it} \\
 & + \sum_{(i,j) \in A_2} \sum_{w \in N_w} \sum_{k \in K} \sum_{t \in T} c_{ij} Y_{wijkt}, \tag{1}
 \end{aligned}$$

Subject to:

$$I_{0t} = I_{0,t-1} + x_t - \sum_{i \in N_c \cup N_w} q_{it} \quad \forall t \in T, \tag{2}$$

$$I_{it} = I_{i,t-1} + q_{it} - d_{it} \quad \forall i \in N_c, \forall t \in T, \tag{3}$$

$$I_{it} = I_{i,t-1} + q_{it} - \sum_{j \in N_c} \sum_{k \in K} r_{ijkt} \quad \forall i \in N_w, \forall t \in T, \tag{4}$$

$$I_{jt} = I_{j,t-1} + \sum_{i \in N_w} \sum_{k \in K} r_{ijkt} - d_{jt} \quad \forall j \in N_c, \forall t \in T, \tag{5}$$

$$I_{it} \leq U_i \quad \forall i \in N, \forall t \in T, \tag{6}$$

$$x_t \leq y_t \times \min\{C, \sum_{t' \in T} \sum_{i \in N_c, N_c} d_{i,t'}\} \quad \forall t \in T, \tag{7}$$

$$q_{it} \leq Q_0 \times v_{it} \quad \forall i \in N_c \cup N_w, \forall t \in T, \tag{8}$$

$$\sum_{j \in N_c} r_{ijkt} \leq Q_i H_{ijkt} \quad \forall i \in N_w, \forall t \in T, \forall k \in K, \tag{9}$$

$$r_{ijkt} \leq H_{ijkt} \times \min \left\{ \sum_{t' \in T} d_{j,t'}, U_j + d_{j,t}, Q_i \right\} \quad \forall i \in N_w, \forall j \in N_c, \forall t \in T, \forall k \in K \tag{10}$$

$$\sum_{i \in N_w} \sum_{k \in K} H_{ijkt} \leq 1 \quad \forall j \in N_c, \forall t \in T, \tag{11}$$

$$L_{wijkt} \leq Q_i Y_{wijkt} \quad \forall w \in N_w, \forall i, j \in N_w \cup N_c, \forall k \in K, \forall t \in T, \tag{12}$$

$$\sum_{j \in N_c} L_{ijkt} = \sum_{j \in N_c} r_{ijkt} \quad \forall i \in N_w, \forall k \in K, \forall t \in T, \tag{13}$$

$$\sum_{i \in N_w \cup N_c} L_{wijkt} - \sum_{i \in N_w \cup N_c} L_{wjikt} = r_{wjkt} \quad \forall w \in N_w, \forall j \in N_c, \forall k \in K, \forall t \in T, \tag{14}$$

$$\sum_{j \in N_w \cup N_{c'}} Y_{wijkt} + \sum_{j \in N_w \cup N_{c'}} Y_{wjikt} = 2H_{wik} \quad \forall w \in N_w, \forall k \in K, \forall t \in T, \quad (15)$$

$$\sum_{j \in N_w \cup N_{c'}, i \neq j} Y_{wijkt} = H_{wik} \quad \forall w \in N_w, \forall i \in N_w \cup N_{c'}, \forall k \in K, \quad \forall t \in T, \quad (16)$$

$$\sum_{i \in N_w \cup N_{c'}} \sum_{j \in N_{c'}} Y_{wijkt} \leq \sum_{j \in N_{c'}} r_{wjkt} / Q_w \quad \forall w \in N_w, \forall k \in K, \forall t \in T, \quad (17)$$

$$x_t \geq 0 \quad \forall t \in T, \quad (18)$$

$$I_{it} \geq 0 \quad \forall i \in N, \forall t \in T, \quad (19)$$

$$q_{it}, v_{it} \geq 0 \quad \forall i \in N_0 \cup N_c \cup N_w, \forall t \in T, \quad (20)$$

$$r_{wjkt} \geq 0 \quad \forall w \in N_w, \forall j \in N_{c'}, \forall k \in K, \forall t \in T, \quad (21)$$

$$L_{wijkt} \geq 0 \quad \forall w \in N_w, \forall i, j \in N_w \cup N_{c'}, \forall k \in K, \quad t \in T, \quad (22)$$

$$y_t \in \{0, 1\} \quad \forall t \in T, \quad (23)$$

$$H_{wjkt} \in \{0, 1\} \quad \forall w \in N_w, \forall j \in N_{c'}, \forall k \in K, \forall t \in T, \quad (24)$$

$$Y_{wijkt} \in \{0, 1\} \quad \forall w \in N_w, \forall i, j \in N_w \cup N_{c'}, \forall k \in K, \quad \forall t \in T. \quad (25)$$

The objective function (1) minimizes the total costs composed of production and setup costs, inventory costs at the manufacturing plant, warehouses, and customers; direct shipment costs for the first echelon; and delivery transportation costs for the second echelon. Constraints (2), (3), (4), and (5) are inventory balance equations at the manufacturing plant, first-level customers, warehouses, and second-level customers, respectively. Constraints (6) ensure that the inventory capacity is respected. Constraint (7) serves two purposes: first, it guarantees that production quantity does not exceed production capacity; second, it links the continuous production variables x_t to the binary setup variables y_t , indicating that a setup is incurred only if there is a positive production level in period t . Constraints (8) and (9) ensure that the total amount delivered by each vehicle to first-level customers and second-level customers, respectively, does not exceed the corresponding vehicle capacity. Constraints (10) establish a link between the amount delivered and the customer visit. Constraints (11) enforce the requirement that each customer must be served by a maximum of one warehouse and one vehicle. According to constraints (12), the load must not exceed the capacity of each vehicle in the second echelon network. Constraints (13) and (14) determine vehicle load by calculating the delivered quantities. Constraints (15), and (16) ensure that a vehicle visits a customer and then leaves after serving them. Constraints (17) impose vehicle capacity requirements and ensure that the vehicle trip arcs selected form a connected graph proposed by Chandra and Fisher (1994). Constraints (13)–(17) are multi-stop adaptations of single-stop routing problem constraints. Finally, constraints (18)–(25) are integrality and non-negativity constraints.

3.1. Numerical example

This section presents an illustrative example of the problem. We assume that a manufacturer has one primary customer and one warehouse. The warehouse can serve two customers within three time periods. The production cost is 30; the setup cost is 3000; and the inventory holding cost is 3

Table 1
Production quantity and setup times

	t1	t2	t3
Production	90	0	0
Setup	1	0	0

Table 2
Inventory quantity

	t1	t2	t3
Manufacturer	0	0	0
Warehouse	13	13	0
Primary customer	20	10	0
Customer 1	14	7	0
Customer 2	13	0	0

Table 3
Delivery quantity of the first level

	t1	t2	t3
Warehouse	60	0	0
Primary customer	30	0	0

Table 4
Delivery quantity of the second level

	t1	t2	t3
Customer 1	21	0	0
Customer 2	26	0	13

for the manufacturer and warehouse. For the primary customer, the holding cost is 6, and for the other two customers, it is 7 and 8, respectively. We assume an initial stock of zero for all nodes. The primary customer demands 10 units for all periods, whereas the other customers require 7 and 13 units. The manufacturer has a production capacity of 100 units per period and an inventory capacity of 50 units. The warehouse has an inventory capacity of 40 units, and each customer has 20 units. The manufacturer can provide retailers with vehicles that have a capacity of 30 units. The warehouse has one vehicle with a 50-unit capacity.

The transportation cost from the manufacturer to the warehouse is 450, and the cost to the primary customer is 162. The transportation costs between the warehouse and the customers (1 and 2) are 385 and 154, respectively, whereas the cost between the two customers is 267.

The optimal solution is 8385, and the detailed solution is presented in Tables 1–4. The manufacturer produces 90 units in the first period, minimizing setup costs. To reduce transportation costs, the manufacturer meets the primary customer's current and future demands by delivering a quantity of 30 units, which results in an inventory of 20 units in the first period and 10 units in the second period, covering the demand for the third period. Additionally, the manufacturer supplies

the warehouse with 60 units from its stock, meeting the demands of Customer 1 in the first period and Customer 2 in both the first and third periods.

4. Solution approach

The 2F-PRP has been proven to be NP-hard, combining two challenging and NP-hard problems: capacitated lot sizing and vehicle routing. As a result, developing exact methods for real-world applications is impractical and not interesting for companies. The latter need solutions to their optimization problems, both for short-term decisions requiring quick responses and long-term decisions exploring numerous scenarios. This section presents an extended version of the two-phase iterative heuristic for classic PRP presented by Absi et al. (2015) followed by our proposed SA heuristic with PR procedure. The effectiveness of both approaches is evaluated by comparing their solutions to those obtained using a commercial solver on the MILP formulation 2E-PRP-MDM.

4.1. Three-phase iterative heuristic approach

As indicated in the review by Adulyasak et al. (2015b), the two-phase iterative algorithm of Absi et al. (2015) stands out as one of the most efficient methods for solving the PRP. This justifies our choice to initially suggest an extension of this algorithm to solve the 2E-PRP-MDM problem. This extension results in the 3PI heuristic presented in this section. The major phases of the heuristic are presented as follows:

- Phase I: *Echelons coordination phase*. This phase incorporates approximate routing costs (SC_{wit}) to determine visit schedules for second-level retailers and delivery quantities r_{wit} while accounting for production capacity, vehicle capacity, and maximum inventory constraints.
- Phase IIa: *Lot sizing with direct shipment phase*. This phase establishes production quantities, visit schedules for warehouses and first-level retailers, delivery quantities, and the required number of vehicles. It takes into consideration production capacity, vehicle capacity constraints, and maximum inventory constraints.
- Phase IIb: *Routing phase*. This phase is designed to optimize routes based on the outcomes of Phase I by solving the vehicle routing problem.

After completing the three phases for a specific number of iterations i_{max} , a diversification mechanism is applied, and this cycle is repeated until the specified iteration limit j_{max} is satisfied. Figure 2 is a flowchart illustrating the general structure of the heuristic.

Further information about the phases and the diversification mechanism are provided in the following subsections.

4.1.1. Phase I: Echelons coordination phase

This phase optimizes the distribution of goods from the supplier to warehouses and also determines when and how to distribute them to the second-level retailers. The cost of visiting second-level customer i by warehouse w in period t is represented by SC_{wit} . The objective is to minimize inventory

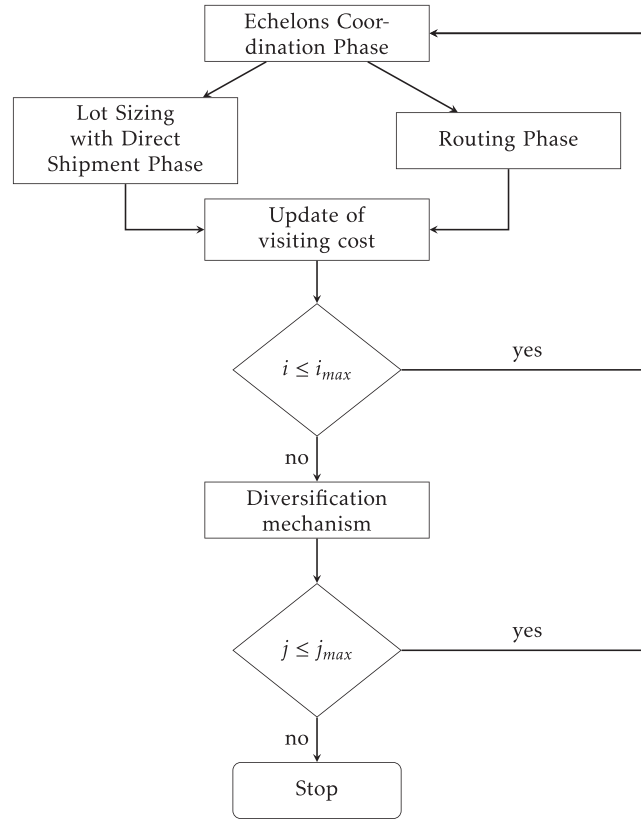


Fig. 2. Flowchart of the proposed 3PI heuristic approach.

costs of warehouses and second-level customers, transportation costs to deliver each warehouse from the manufacturing plant, as well as costs related to inserting second-level customers into vehicle routes.

The first phase model (PhI-MILP) is

$$[\mathbf{PhI} - \mathbf{MILP}] \quad \min \sum_{i \in N_w \cup N_{c'}} \sum_{t \in T} h_i I_{it} + \sum_{t \in T} \sum_{i \in N_w} c_i v_{it} + \sum_{w \in N_w} \sum_{i \in N_{c'}} \sum_{t \in T} SC_{wit} H_{wit} \quad (26)$$

Subject to Constraints (4) – (6), and

$$q_{it} \leq Q_0 \times v_{it} \quad \forall i \in N_w, \forall t \in T, \quad (27)$$

$$\sum_{j \in N_{c'}} r_{ijt} \leq \lambda_{it} \times |K| \times Q_i \quad \forall i \in N_w, \forall t \in T, \forall k \in K, \quad (28)$$

$$r_{ijt} \leq H_{ijt} \times \min \left\{ \sum_{t' \in T} d_{j,t'}, U_j + d_{j,t}, Q_i \right\} \quad \forall i \in N_w, \forall j \in N_{c'}, \forall t \in T, \quad (29)$$

$$\sum_{i \in N_w} H_{ijt} \leq 1 \quad \forall j \in N_c, \forall t \in T, \tag{30}$$

$$\sum_{i \in N_w} q_{it} + \sum_{i \in N_c} \delta_{it} \leq C \quad \forall t \in T, \tag{31}$$

$$I_{it} \geq 0 \quad \forall i \in N_w \cup N_c, t \in T, \tag{32}$$

$$q_{it}, v_{it} \geq 0 \quad \forall i \in N_w, t \in T, \tag{33}$$

$$r_{ijt} \geq 0 \quad \forall i \in N_w, \forall j \in N_c, t \in T, \tag{34}$$

$$H_{ijt} \in \{0, 1\} \quad \forall i \in N_w, \forall j \in N_c, \forall t \in T. \tag{35}$$

Constraints (28) indicate that the delivery quantities fulfill the total capacity of the vehicles available at each time point. These constraints use a factor λ_{it} which is a scaling parameter that starts at 1 and gets adjusted later based on specific rules that will be clarified at the end of this section. Constraints (29) and (30) play the same role as constraints (10) and (11) in the 2E-PRP-MDM with removing vehicle index k from the r and H variables. In constraints (31), $\delta_{it} = \max(0, d_{it} - \max(0, I_{i0} - \sum_{t'=1}^{t-1} d_{it'}))$, $\forall i \in N_c$, at the beginning. Then δ_{it} will be updated based on the delivery quantity in the first echelon. Production capacity is ensured by these constraints.

4.1.2. Phase IIa: Lot sizing with direct shipment

This phase is responsible for determining the production quantities, visit schedules for warehouses and first-level retailers, delivery quantities, and the required number of vehicles. It takes into account the production, maximum inventory, and vehicle capacity constraints. The goal of this phase is to ensure that the production process is optimized and that the products are delivered to the customers in a timely and efficient manner. By considering the production capacity, vehicle capacity constraints, and maximum inventory constraints.

The objective of this phase is to minimize production, inventory costs, and transportation costs. The second phase model (PhIIa-MILP) is as follows:

$$[\text{PhIIa} - \text{MILP}] \quad \min \sum_{t \in T} px_t + \sum_{t \in T} fy_t + \sum_{i \in N_0 \cup N_c \cup N_w} \sum_{t \in T} h_i I_{it} + \sum_{t \in T} \sum_{i \in N_c \cup N_w} c_i v_{it} \tag{36}$$

Subject to Constraints (2), (3), (6), (8), (18) – (20), (23), and

$$x_t \leq y_t \min \left\{ C, \sum_{t' \in T} \sum_{i \in N_c, N_w} d_{i,t'} \right\} \quad \forall t \in T. \tag{37}$$

In constraints (37), the total amount of goods delivered from a warehouse to its served customers is considered as its demand, $d_{it} = \sum_{j \in N_c} r_{ijt}$, $\forall i \in N_w$.

4.1.3. Phase IIb: Routing phase

The decision variables from Phase I, r_{ijt} and H_{ijt} , are used in Phase IIb to solve a vehicle routing problem (VRP) and a traveling salesman problem (TSP) in the case of $K \geq 2$ at each period. A heuristic approach is used to solve the VRP and TSP in order to reduce computing time. We

Table 5
Visiting Cost SC_{wit}

	t1	t2	t3
C1	10	10	10
C2	30	30	30
C3	5	5	5
C4	15	15	15

Table 6
Visiting periods H_{wit}

	t1	t2	t3
C1	0	1	1
C2	1	1	0
C3	0	1	0
C4	1	1	0

have used the LKH-3 implementation from Helsgaun (2017). For each warehouse $i \in N_w$, when the LKH-3 heuristic fails to find a feasible solution for a period t , the value of λ_{it} is decreased: $\lambda_{it} = \lambda_{it} - \eta$, where η is a constant. Conversely, when LKH-3 finds a feasible solution, the value of λ_{it} is incremented unless it already equals 1, $\lambda_{it} = \min(\lambda_{it} + \eta, 1)$. The diversification involves setting all λ_{it} values back to their initial state of 1.

4.1.4. Update of visiting cost

Visiting costs are used to create a connection between Phase I and Phase IIb of the approach. They are initially set based on the transportation costs and dynamically updated based on Phase IIb solutions to improve retailer clustering in subsequent iterations.

First, we define some parameters. Let S_{wt} be the set of all retailers that are served during time period t by warehouse w . Let i^- be the predecessor of vertex i and i^+ its successor. For $k \in K$, $t \in T$, and $i \notin S_{wt}$, let Δ_{kit} be the cheapest insertion cost for inserting i in the route of vehicle k . For each warehouse $w \in N_w$, $t \in T$ and $k \in K$ when $i \in S_{wt}$, $SC_{wit} = c_{i^-i} + c_{ii^+} - c_{i^-i^+}$. Conversely, when $i \notin S_{wt}$, $SC_{wit} = \min_{k \in K} \Delta_{kit}$.

4.1.5. Diversification mechanism

The diversification mechanism employs the best-known solution to modify “visiting costs,” guiding the heuristic search toward unexplored regions of the solution space. It achieves this by multiplying the visiting cost SC_{wit} of each retailer in period t by the number of retailers served in that period plus one, leveraging information from the current best solution. The diversification mechanism encourages retailers to shift from high to low numbers of visits. The following example explains the mechanism.

We have one warehouse, four retailers, and a planning horizon of three time periods. The visiting cost SC_{wit} and the visiting time H_{wit} are shown in Tables 5 and 6.

Table 7
New Visiting Cost SC_{wit}

	t1	t2	t3
C1	21	41	11
C2	61	121	31
C3	11	21	6
C4	31	61	16

Table 8
Solution 1

	t1	t2	t3
C1	1	0	1
C2	1	1	1
C3	0	1	1
C4	1	1	1

Table 9
Solution 2

	t1	t2	t3
C1	1	0	1
C2	1	0	0
C3	1	0	0
C4	1	0	0

From Table 6 and the values of H_{wit} , we conclude the number of retailers visited in each period: 2, 4, 1. We multiply the latter values by the visiting cost plus 1. The updated SC_{wit} is shown in Table 7. The new SC_{wit} can help explore the search space by shifting some retailers from periods with high visiting times to lower ones. Tables 8 and 9 present two examples of solutions. Solution 1 recommends visiting the first retailer in the first period because of the lower costs. Solution 2 suggests moving all retailer visits from the second period to the first period because of the lower costs in the first period for all retailers.

4.2. SA-PR procedure

The SA-PR algorithm is a heuristic composed of three steps.¹ Step I involves generating the initial solution. In Step II, SA is used to refine the initial solution. Then, the γ best solutions from this process are chosen for further improvement. Finally, Step III is dedicated to enhancing the chosen solution through the application of the PR technique.

¹We deliberately avoided the use of the word *Phase* to avoid confusion with the phases of the 3PI heuristic.

4.2.1. Step I: Initial solution

This step consists of the first iteration in the 3PI heuristic described in Section 4.1. Hence it corresponds to running Phase I, Phase IIa, and Phase IIb only once.

This resulted in the construction of an initial solution.

4.2.2. Step II: Simulated annealing

The initial solution obtained in Step I is improved using the SA meta-heuristic. The SA algorithm was introduced by Kirkpatrick (1984). It is inspired by the metallurgical process of annealing, in which a material is gradually cooled to create a low-energy condition. The SA algorithm is known for its simplicity and its effectiveness in addressing various optimization problems in logistics and production systems (Koulamas et al., 1994; Ozsen et al., 2020), inventory routing problems (Alvarez et al., 2018), routing problems (Ceschia et al., 2026; Gündüz Mengübaşı et al., 2025), and in the TSP (Zhou et al., 2022). (See Delahaye et al., 2019, for a recent review of advances and applications of SA.)

One major difference between SA and other metaheuristics is the fact that, at each iteration, it generates only one neighbor for the current solution. If the newly created solution is better than the current one, it takes its position. If not, there is a chance it will be agreed. Our SA algorithm consists of three phases. In the first phase, new distribution plans are generated for both echelons. The second phase is dedicated to establishing the production schedule, and the third phase focuses on determining the routing for vehicles. More details are given below.

1. *Delivery neighboring generation*: We obtain from the initial solution q_{it} and r_{ijt} . These data are used for generating neighbors by transferring the delivery quantity $s_{it,t'}$ from company to node $i \in N_c \cup N_w$ from period t to period t' . We start with the first echelon which contains the company, first-level customers, and warehouses:

Select randomly a set of nodes E (first-level customers and warehouses), two periods t and t' :
for $i \in E$: if $t' < t$:

$$s_{it,t'} = \min \left\{ q_{it}, \left(t' \times C - \sum_{i \in N_c \cup N_w} \sum_{t'' \in t'} q_{i,t''} \right), \min_{t''} \{ U_i - I_{i,t''}, I_{0,t''} \} \right\}, \quad (38)$$

and if $t' > t$:

$$s_{it,t'} = \min \left\{ q_{it}, \min_{t''} \{ I_{i,t''} \} \right\}. \quad (39)$$

For the second echelon, we make the transfer for r_{ijt} as follows:

Randomly select a group of second-level customers (E) alongside two warehouses (w_1 and w_2) and two periods (t and t'). These selections define potential neighbor structure by transferring delivery quantities ($s_{i,w_1,w_2,t,t'}$) for each customer ($i \in E$), moving them from warehouse w_1 in period t to warehouse w_2 in period t' :

for $i \in E$: if $t' < t$:

$$s_{i,w_1,w_2,t,t'} = \min \left\{ r_{w_1,it}, \left(t' \times C - \sum_{j \in N_c \cup N_w} \sum_{t'' \in t'} q_{j,t''} \right), \left(Q_{w_2} \times K - \sum_{j \in N_c} r_{w_2,j,t'} \right), \min_{t''} \{ U_i - I_{i,t''}, I_{w_2,t''} \} \right\}, \tag{40}$$

and if $t' > t$

$$s_{i,w_1,w_2,t,t'} = \min \left\{ r_{w_1,it}, \left(Q_{w_2} \times K - \sum_{j \in N_c} r_{w_2,j,t'} \right), \min_{t''} \{ I_{i,t''} \} \right\}. \tag{41}$$

Since split delivery is not allowed, the transfer from period t to t' when $w_1 \neq w_2$ and $s_{i,w_1,w_2,t,t'} < r_{w_1,it}$ will be 0. After each transfer, we have to update the delivery quantity from the company to each warehouse w_1 and w_2 . The neighborhood procedure described in this paper draws inspiration from the research conducted by Armentano et al. (2011). We recommend that readers refer to that paper for more details.

2. *Lot sizing model*: In this phase, we must adapt the production planning to the new delivery planning q_{it} for the first echelon received from Phase I. The production planning problem corresponds to a lot-sizing problem, which is addressed using a standard MILP solver. The LSP model is as follows:

$$[\mathbf{LSP} - \mathbf{M}] \quad \min \sum_{t \in T} p x_t + \sum_{t \in T} f y_t + \sum_{t \in T} h_0 I_{0,t} \tag{42}$$

subject to (2), (6), (7), (18), (23), and (43)

$$x_t \leq y_t \sum_{t' \in T} \sum_{i \in N_c, N_w} q_{i,t'} \quad \forall t \in T. \tag{44}$$

3. *Routing model*: In this phase, we define the routes of vehicles using the new delivery plans q_{it} and r_{ijt} . For the first echelon, we use q_{it} to define the number of vehicles needed (v_{it}) to serve each customer and warehouse, and then calculate the transportation cost $\sum_{t \in T} \sum_{i \in N_c \cup N_w} c_i v_{it}$. To compute the routing cost $\sum_{(i,j) \in A_2} \sum_{w \in N_w} \sum_{k \in K} \sum_{t \in T} c_{ij} Y_{wijkt}$ in the second echelon, we use r_{ijt} and the LKH-3 implementation of Helsgaun (2017) to solve the VRP and TSP for each period.

4.2.3. Step III: Path relinking

PR is a technique that explores neighboring solutions along a hypothetical path in the search space, connecting two solutions. Glover (1997) introduced PR, a method combining both *intensification* and *diversification* aspects within tabu search and scatter search algorithms. It has been successfully used by Ho and Gendreau (2006) and Peng et al. (2015) in tabu search and by Máximo and Nascimento (2021) in adaptive iterated local search to solve vehicle routing and operations scheduling problems. It has been combined with GRASP by Interian and Ribeiro (2017) and Cuellar-Usaquén

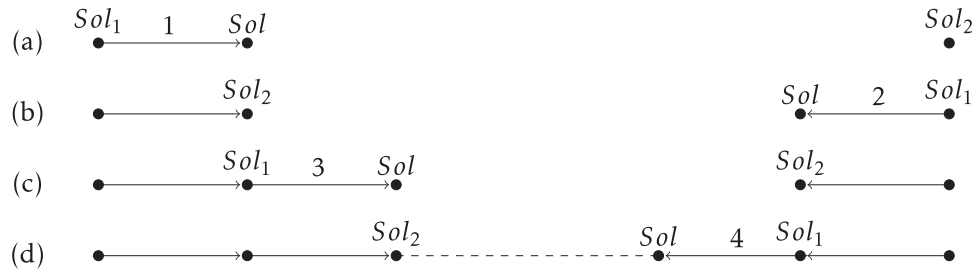


Fig. 3. Mixed path relinking.

et al. (2023) to solve the TSP. In the domain of combinatorial optimization, PR facilitates the exploration of trajectories connecting elite solutions, which represent high-quality outcomes. This exploration utilizes the search space graph, denoted as $G = (F, M)$, where nodes correspond to the set F of feasible solutions. Typically, PR operates between two solutions within F . One serves as the initial solution, Sol_1 , while the other acts as the guiding (target) solution, Sol_2 . Through PR, one or more paths connecting these solutions within the search space graph can be investigated to locate superior solutions. Notably, local search is often applied to the best solution encountered on each explored path, as it might not represent the optimal solution within its immediate neighborhood. PR has various implementation strategies. These consist of not only forward PR but also backward and mixed PR.

We have used mixed PR which is applied between two feasible solutions, Sol_1 and Sol_2 , to explore the connecting path bidirectionally. In each iteration, the search alternates between the original initial solution Sol_1 and the guiding solution Sol_2 . This creates the impression of traversing two subpaths, one from Sol_1 to Sol_2 and the other from Sol_2 to Sol_1 . The subpaths converge at a feasible solution midway, establishing a single path that connects Sol_1 and Sol_2 . Figure 3 depicts the process of applying mixed PR to two solutions, Sol_1 and Sol_2 .

We have defined the set E of elite solutions, containing the γ best solutions identified throughout the SA search. The solutions in E must have a minimum distance between them in order to maintain a degree of diversity. The distance between two solutions is defined as the number of periods in which the amounts delivered to customers differ. New solutions are added to E if they have a smaller cost than the worst solution in E , or if they have a smaller cost than the best solution in E and increase the distance between solutions in E . In either case, the worst solution in E is replaced. We have selected the initial solution Sol_1 as the SA solution, while the guide solution Sol_2 is an E solution with the maximum distance $D_{Sol_2}^{Sol_1}$ from Sol_1 . This ensures that the solutions in E are diverse and that the PR process does not get stuck in a local optimum.

The PR algorithm for the first echelon is inspired by Armentano et al. (2011) and presented in Algorithm 1. It starts with two solutions, Sol_1 and Sol_2 . The algorithm starts with two solutions, Sol_1 and Sol_2 (line 1), and iteratively transfers quantities from one solution to another, comparing the delivery quantities $q_{it}^{Sol_1} \neq q_{it}^{Sol_2}$ for each warehouse and first-level customer and period in the two solutions (lines 3–5). For each period $t' \neq t$, the algorithm makes the transfer from period t to period t' in solution Sol_1 , with respect to production and inventory capacities (lines 6–11). The algorithm then calculates the new solution's total cost (line 12). The best solution is saved as Sol_3 (line 16). If the distance between Sol_2 and Sol_3 is not increased, then the algorithm sets Sol_1 to Sol_2 and Sol_2 to Sol_3 (lines 17–19). This process is repeated until Sol_1 and Sol_2 are the same.

Algorithm 1. Path Relinking

```

1: Input:  $Sol_1, Sol_2, D_{Sol_2}^{Sol_1}$ 
2: while  $Sol_1 \neq Sol_2$  do
3:   for  $i \in N_c \cup N_w, t \in T$  do
4:     if  $q_{it}^{Sol_1} \neq q_{it}^{Sol_2}$  then
5:        $\Theta_{it} = q_{it}^{Sol_1} - q_{it}^{Sol_2}$ 
6:       for  $t' \neq t$  do
7:         if  $t' \leq t$  then
8:            $s_{it,t'} = \min\{|\Theta_{it}|, (t' \times C - \sum_{i \in N_c \cup N_w} \sum_{t'' \in t'} q_{i,t''}^{Sol_1}), \min_{t''} U_i - I_{i,t''}, I_{0,t''}\}$ 
9:         else
10:           $s_{it,t'} = \min\{|\Theta_{it}|, \min_{t''}\{I_{i,t''}\}\}$ 
11:        end if
12:       Determine the total cost after transferring  $s_{it,t'}$ 
13:     end for
14:   end if
15: end for
16:  $Sol_3$  is the best solution found after the transfer  $s_{it,t'}$ 
17: if  $D_{Sol_3}^{Sol_2} \leq D_{Sol_2}^{Sol_1}$  then
18:    $Sol_1 = Sol_2$ 
19:    $Sol_2 = Sol_3$ 
20: end if
21: end while

```

The PR of the second echelon is the same as Algorithm 1 with some changes.

It starts with two solutions, Sol_1 and Sol_2 (line 1), and iteratively transfers quantities from one solution to another, comparing the delivery quantities $r_{wit}^{Sol_1} \neq r_{wit}^{Sol_2}$ from each warehouse $w \in N_w$ for each second-level customer $i \in N_c$ and period $t \in T$ in the two solutions (lines 3 and 4) the equation in line 5 is $\Theta_{wit} = r_{wit}^{Sol_1} - r_{wit}^{Sol_2}$. For each period $t' \neq t$, the algorithm shifts from period t to period t' and from warehouse w to warehouse w' in solution Sol_1 , with respect to production and inventory capacities (lines 6–11). In line 8, $s_{i,w,w',t,t'} = \min\{|\Theta_{wit}|, (t' \times C - \sum_{j \in N_c \cup N_w} \sum_{t'' \in t'} q_{j,t''}), (Q_{w'} \times K - \sum_{j \in N_c} r_{w',j,t'}), \min_{t''}\{U_i - I_{i,t''}, I_{w',t''}\}\}$. In line 10, $s_{i,w,w',t,t'} = \min\{|\Theta_{wit}|, (Q_{w'} \times K - \sum_{j \in N_c} r_{w',j,t'}), \min_{t''}\{I_{i,t''}\}\}$. Finally, determine the total cost of the new solution (line 12). The best solution is saved as Sol_3 (line 16). If the distance between Sol_2 and Sol_3 is not increased, the algorithm sets Sol_1 to Sol_2 and Sol_2 to Sol_3 (lines 17–19). This process is repeated until Sol_1 and Sol_2 are the same.

4.2.4. Pseudo-code of SA with path relinking

In this context, we outline the integration of SA-PR within our implementation. The resulting procedure is described in Algorithm 2. The algorithm begins by initializing an empty set, E , which will be used to store solutions. An initial solution, Sol , is constructed as described in Section 4.2.1. The algorithm then executes SA-PR for a limited number of iterations. The SA procedure described in lines 6 to 24 is as follows. The algorithm enters a loop that repeats until the temperature τ drops below the final temperature τ_f . The current solution denoted as Sol is altered to create a new neighboring solution represented by Sol' as described in Section 4.2.2. The cost difference

Algorithm 2. Simulated Annealing with Path Relinking

```

1: set  $\tau_0, \tau_f, \mu$ 
2:  $E = \emptyset$ 
3: Let  $Sol$  be the initial solution.
4: Set  $Sol_{best} = Sol$ 
5: while a stopping criterion is not met do
6:    $\tau = \tau_0$ 
7:   while  $\tau > \tau_f$  do
8:     Generate Neighbor solution  $Sol'$  for  $Sol$  as follows: use Phase I, Phase II, and then Phase III
9:      $\Delta = cost(Sol') - Cost(Sol)$ 
10:    if  $\Delta \leq 0$  then
11:       $Sol = Sol'$ 
12:    else
13:       $a = Uniform[0, 1]$ 
14:       $\alpha = e^{(-\Delta/\tau)}$ 
15:      if  $a \leq \alpha$  then
16:         $Sol = Sol'$ 
17:      end if
18:    end if
19:    if  $Sol$  is not in  $E$  and respects conditions it enters  $E$ ,  $E = E \cup S$ 
20:    if  $cost(Sol) \leq cost(Sol_{best})$  then
21:       $Sol_{best} = Sol$ 
22:    end if
23:     $\tau = \tau \times \mu$ 
24:  end while
25:  $Sol_1 = Sol_{best}$ 
26: Set  $Sol_2$  as the solution with maximum distance  $D$  from  $Sol_{best}$  in  $E$ 
27:  $Sol = Path\ Relinking(Sol_1, Sol_2, D_{Sol_1}^{Sol_1})$  of the first echelon
28: if  $cost(Sol) \leq cost(Sol_{best})$  then
29:    $Sol_{best} = Sol$ 
30: end if
31:  $Sol = Path\ Relinking\ Sol, Sol_2, D_{Sol_2}^{Sol}$  of the second echelon
32: if  $cost(Sol) \leq cost(Sol_{best})$  then
33:    $Sol_{best} = S$ 
34: end if
35: end while

```

between the new solution Sol' and the current solution Sol is calculated as Δ . If the cost difference Δ is negative, the new solution Sol' is accepted as the current and will replace Sol . Otherwise, the new solution Sol' may still be accepted with a certain probability α . The solution Sol enters the set E if it does not already exist in E and meets the conditions presented in Section 4.2.3. If the cost of the current solution Sol is lower than that of the best solution, then Sol becomes the new best solution. Subsequently, the temperature τ is reduced by a factor of μ . After the SA algorithm is complete, the set E is filled with solutions. We select the initial solution Sol_1 as the best solution found in SA, and the guiding solution Sol_2 as the solution farthest from Sol_1 . Lines 27–33 execute PR for both echelons. SA will continue with the solution found by PR as the current solution.

5. Numerical experiments

This section has two main objectives. First, we will assess the computational performance of the proposed heuristics: The 3PI heuristic and the SA-PR heuristic. Second, we will discuss our results from a managerial perspective. The MILP formulation 2E-PRP-MDM was implemented and solved on CPLEX 12.8 using its default configuration. The 3PI and SA-PR heuristics were implemented in Python 3.8 on Jupyter Notebook.

All experiments were carried out on a 64-bit Windows 11 with Intel(R) Xeon(R) CPU E5-2673 v4 @ 2.30 GHz and 14 GB RAM. The 3PI heuristic employs a diversification mechanism every five iterations of the three phases. The heuristic is terminated after a total of 100 iterations. The first and second phases have a 100-second time limit each. The SA-PR parameters τ_0 , τ_f , and μ were set to 140, 1, and 0.9, respectively. This setting was inspired by Shaabani and Kamalabadi (2016). We have configured the SA-PR heuristic to complete three iterations. We have noticed that more iterations do not significantly improve the solution while the CPU increases considerably. The lot sizing decisions (model *LSP-M*) are limited to 100 seconds of CPU time.

5.1. Generated datasets

Our problem instances were created by modifying benchmarks from Archetti et al. (2011) and using data settings from Adulyasak et al. (2014a). Archetti et al. (2011) presented these instances and are accessible on their website (<http://orbrescia.unibs.it/instances>). The cited paper introduces four variations of PRP instances. The first is the base case, the second increases production costs, the third increases transportation costs, and the fourth removes holding costs for retailers altogether. Each class has 19, 50, and 100 retailers. This algorithm is designed for a specific type of PRP problem, where the product demand is constant and the manufacturing plant has unlimited capacity. We generated our dataset using instances that have 50 customers. We used (i, j) customers for the first and second echelons equal to (2,3), (4,6), (6,12), and (10,22) for each set of warehouse $W = \{1, 2, 3\}$, and time horizons with $|T| = 3$ and 6. The number of vehicles is designated as $|K| = 1$ or 2 for instances where $j \leq 12$, and $|K| = 2$ or 3 where $j \geq 22$. We have assumed that a given customer's demand for a given period does not exceed the vehicle's capacity. This assumption ensures that each customer is served by a single warehouse and a single vehicle as in the MIPL model. Several similar studies have made this assumption; see, for example, Senoussi et al. (2016) and Absi et al. (2018). Allowing split deliveries would require modifications to the model, significantly increasing its complexity.

As the *base case*, we chose instances from the first class, while instances from the second class were selected as cases where production costs are high called the *high production case*. Likewise, instances from the third class were chosen when transportation costs were high called the *high transportation case*. Additionally, for instances involving low inventory holding costs, we opted for instances from the first class when inventory costs are lower than the base case, which we called the *low inventory case*. As a result, we obtain $4 \times 3 \times 2 \times 2 \times 4 = 192$ instances to test the algorithm's performance. All used instances and detailed results of these simulations are available upon request from the authors.

5.2. Performance analysis

In the first experiment, we assessed the performance of our heuristics compared to the optimal solution (when available) or the best upper bound identified by CPLEX after a 3600-second time limit. To analyze the results effectively, we categorized the instances into three groups based on their total number of customers $N_T = N_c + N_e$ where small-size instances include all cases with 6 and 11 customers, medium-size instances with 19 customers, and large-size instances with 33 customers. In the summary tables, we organize the instances into groups based on various factors: the specific cases, the number of warehouses (N_w), and the planning horizon (T). Each table is divided into three sections: 3PI heuristic, SA-PR, and CPLEX CPU time. Both heuristic sections contain two main columns. The first column corresponds to CPU times, where CPU^3 represents the CPU time of the 3PI heuristic, and CPU^s represents the CPU time of the SA-PR. Additionally, the columns labeled GAP^3 and GAP^s calculate the average percentage gap for each heuristic. This gap is computed as follows:

$$GAP^3 = 100 \times \frac{Sol^3 - UB^{CPLEX}}{UB^{CPLEX}}, \quad (45)$$

$$GAP^s = 100 \times \frac{Sol^s - UB^{CPLEX}}{UB^{CPLEX}}, \quad (46)$$

where Sol^3 and Sol^s represent the solution produced by the 3PI and SA-PR, respectively, and UB^{CPLEX} represents the upper bound value. The final section provides information about the computational time taken by CPLEX. Note that the values indicated in bold in Tables 10–12 indicate the heuristic that achieves the lowest gap.

Small instance analyses are summarized in Table 10. Notably, the SA-PR consistently produces high-quality solutions, closely approaching the solution value obtained by CPLEX. On average, the gap between SA-PR solutions and CPLEX solutions is only 0.87%, outperforming the 3PI heuristic in 87% of cases. The average gap values (GAP^s) vary from 0.08% to 3.14% for SA-PR. On the other hand, 3PI can find solutions close to the solution of CPLEX with a gap of only 0.1% (specifically observed in cases involving high production, two warehouses, and six periods). When considering computational time, the 3PI algorithm generally completes computations quicker than the SA-PR algorithm. However, both methods require more time compared to the CPLEX solver.

In the context of medium-sized problems, both proposed heuristics demonstrate exceptional performance, as shown in Table 11. These heuristics consistently yield high-quality solutions, with average gaps of only 2.64% for SA-PR and 3.75% for the 3PI heuristic when compared to the solution obtained from CPLEX. Additionally, in terms of computational time, SA-PR and 3PI outperform CPLEX significantly. On average, SA-PR completes its computations in 61.48 seconds—10 times faster than CPLEX. Remarkably, 3PI is even quicker, averaging 31.31 seconds, which is 20 times faster than CPLEX. The strong results highlight the heuristics' talent for rapid discovery of quality solutions, even under varied problem configurations.

Table 12 presents the results obtained for large-sized instances. In Table 12, we observe that both heuristics yield high-quality solutions, with an average gap of 2.38% and 2.66% compared to the solution obtained by CPLEX for the 3PI heuristic and SA-PR, respectively. Notably, both heuristics

Table 10
Performance of the proposed heuristics against CPLEX for small-size instances

Cases	N_w	T	3PI		SA-PR		CPU ^{CPLEX}
			CPU ³	GAP ³	CPU ^s	GAP ^s	
Base case	1	3	6.00	2.88	15.25	1.55	0.31
		6	11.00	0.84	20.00	0.89	1.01
	2	3	9.25	3.51	16.00	1.95	1.36
		6	16.25	0.75	20.75	0.74	8.39
	3	3	9.75	2.62	18.00	1.63	5.91
		6	19.50	1.53	22.00	1.20	10.25
High production cost	1	3	6.00	0.40	15.25	0.20	0.49
		6	12.00	0.12	20.75	0.12	1.82
	2	3	8.25	0.49	23.75	0.23	2.09
		6	17.00	0.10	20.50	0.08	5.72
	3	3	10.00	0.36	18.50	0.23	3.85
		6	20.00	0.21	21.00	0.21	10.43
High transportation cost	1	3	5.75	4.33	13.50	1.51	0.41
		6	16.00	3.07	43.75	1.21	3.83
	2	3	7.50	9.85	11.50	1.13	1.04
		6	15.00	4.12	38.50	1.43	38.30
	3	3	8.50	4.24	12.00	3.14	2.77
		6	19.50	5.14	17.00	0.93	37.23
Low inventory cost	1	3	6.50	1.37	14.75	0.19	0.35
		6	11.75	0.31	22.50	0.60	2.22
	2	3	9.25	1.64	18.50	0.19	1.04
		6	17.25	1.25	32.00	0.49	5.80
	3	3	9.00	2.52	12.25	0.45	2.50
		6	18.50	1.26	21.25	0.64	11.74
Average			12.06	2.20	20.39	0.87	6.62

achieve a minimum gap of 0.06% (in the case of high production costs with three warehouses and three periods).

Regarding CPU time, the SA-PR and 3PI heuristics significantly outperform CPLEX. On average, the heuristics require only 206.29 and 212.33 seconds, respectively—16 times less than the CPU time used by CPLEX for SA-PR and 3PI. This impressive performance demonstrates that the heuristics swiftly identify high-quality solutions, regardless of the instance settings.

5.3. Computational performance of heuristics on medium- and large-sized instances

To ensure a fair comparison of performance between CPLEX and the two heuristics (SA-PR and 3PI heuristic), the CPLEX time limit has been adjusted for medium- and large-sized instances. This adjustment involves using the average and maximum CPU times recorded by the heuristics when solving instances in those size categories. Tables 13 and 14 present the findings from this analysis. The table is split into two sections based on CPLEX's time limits: one for 45 seconds and the other for 75 seconds for medium-sized instances and 200 and 450 seconds for large-sized instances. Each section contains two columns: GAP^3 and GAP^s for 3PI heuristic and SA-PR. These values are calculated using Equations (45) and (46).

Table 11
Performance of the proposed heuristics against CPLEX for medium-size instances

Cases	N_W	T	3PI	GAP ³	SA-PR	GAP ^s	CPU ^{CPLEX}
			CPU ³		CPU ^s		
Base case	1	3	14.50	2.13	47.50	1.22	7.21
		6	25.50	0.82	52.00	1.26	107.4
	2	3	18.50	2.92	126.50	2.40	46.11
		6	32.50	1.41	50.00	1.17	1839.10
	3	3	22.00	2.93	43.00	3.01	170.98
		6	44.50	3.18	46.50	3.83	1936.55
High production cost	1	3	14.00	0.31	53.00	0.14	6.13
		6	25.50	0.13	64.50	0.14	312.97
	2	3	18.50	0.43	42.00	0.41	44.94
		6	36.00	0.22	51.00	0.20	1847.73
	3	3	21.00	0.38	53.00	0.37	231.42
		6	47.00	0.45	42.50	0.56	2121.13
High transportation cost	1	3	11.00	9.77	52.00	1.99	8.89
		6	50.50	7.85	32.00	6.71	318.41
	2	3	15.00	8.29	29.00	9.06	27.63
		6	49.50	4.73	29.00	5.64	1840.50
	3	3	19.50	11.41	27.00	3.14	244.91
		6	75.00	11.47	29.00	6.92	1979.00
Low inventory cost	1	3	13.50	2.56	129.00	1.08	7.19
		6	46.00	1.39	210.50	1.72	145.71
	2	3	60.50	2.86	87.00	2.76	1388.73
		6	18.50	4.48	61.00	1.29	37.98
	3	3	20.50	5.20	71.50	3.56	164.95
		6	52.50	4.67	47.00	4.67	1848.95
Average			31.31	3.75	61.48	2.64	695.19

In Table 13, both heuristics exhibit commendable performance. For the 75-second time limit, the average gap is 0.27% for the 3PI heuristic and -0.5% for SA-PR. Similarly, for the 45-second time limit, the gaps are -0.14% and -0.9% , respectively. Notably, both heuristics demonstrate improvement of up to 24%. These findings suggest that SA-PR surpasses both 3PI and CPLEX across both time limits. However, 3PI outperforms CPLEX only within the 45-second time limit.

Table 14 illustrates the superior performance of both heuristics over CPLEX. Specifically, when the time limit is set to 200 seconds, the average improvement is -5.23% for the 3PI heuristic and -4.97% for SA-PR. Similarly, with a 450-second time limit, the improvements are -4.47% and -4.2% , respectively. These heuristics exhibit the capability to enhance results by up to 31% compared to CPLEX, across approximately 60% of instances.

5.4. Results analysis

In this study, we conducted an analysis focusing on the performance of two heuristics 3PI and SA-PR across four different cases: base case, high production cost, high transportation cost, and low inventory cost which is represented by Figs. 4 and 5.

Table 12
Performance of the proposed heuristics against CPLEX for large-size instances

Cases	N_w	T	3PI		SA-PR		CPU ^{CPLEX}
			CPU ³	GAP ³	CPU ^s	GAP ^s	
Base case	1	3	292.00	1.57	164.50	1.58	1828.28
		6	459.50	0.96	600.00	1.80	3600.00
	2	3	75.00	1.64	139.50	1.47	3600.00
		6	414.50	0.40	617.50	1.37	3600.00
	3	3	107.00	1.27	101.00	1.23	3600.00
		6	267.00	1.58	80.00	1.92	3600.0
High production cost	1	3	291.50	0.24	159.00	0.23	1499.54
		6	463.00	0.15	670.50	0.28	3600.00
	2	3	77.00	0.24	149.00	0.21	3283.21
		6	415.50	0.10	609.00	0.27	3600.00
	3	3	229.50	0.06	105.50	0.06	3600.00
		6	353.50	0.23	76.50	0.28	3600.00
High transportation cost	1	3	22.00	6.03	166.00	5.63	3600.00
		6	201.00	3.30	107.50	5.25	3600.00
	2	3	54.00	5.29	67.00	7.15	3600.00
		6	174.50	3.49	96.00	5.07	3600.00
	3	3	68.50	4.56	63.50	2.47	3600.00
		6	189.00	6.25	85.50	6.01	3600.00
Low inventory cost	1	3	42.50	3.64	208.00	3.26	2843.34
		6	245.00	2.69	149.50	3.24	3600.00
	2	3	82.50	4.54	192.50	3.92	2257.42
		6	267.50	1.95	118.50	3.03	3600.00
	3	3	82.50	3.75	151.50	4.66	3600.00
		6	222.00	3.22	73.50	3.39	3600.00
Average			212.33	2.38	206.29	2.66	3337.99

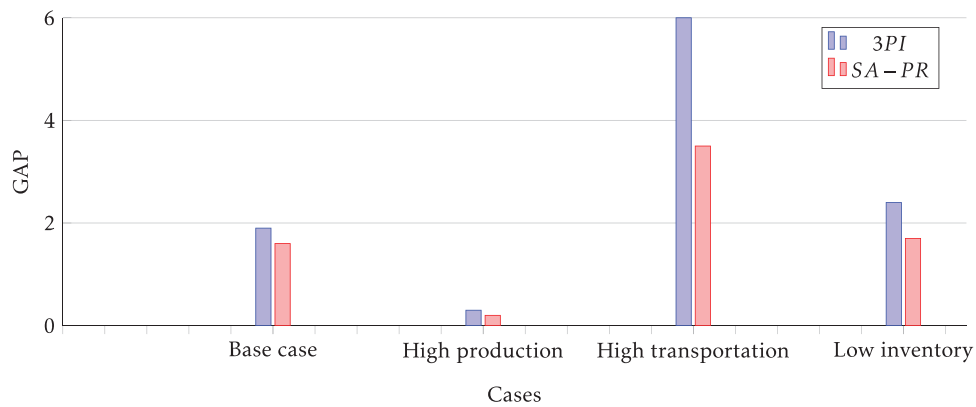


Fig. 4. Comparison of the average gap between 3PI and SA-PR heuristics for all cases.

Table 13

Performance comparison of proposed heuristics and CPLEX with 45 and 75 seconds time limits for medium-sized instance

Cases	N_w	T	45 seconds		75 seconds		
			GAP ³	GAP ^s	GAP ³	GAP ^s	
Base case	1	6	−24.60	−24.27	−23.41	−23.07	
	2	3	2.58	2.06	2.92	2.40	
	3	6	0.89	0.65	1.28	1.04	
		3	3	1.20	1.28	1.86	1.94
		6	1.55	2.19	1.99	2.64	
High production cost	1	6	−18.74	−18.73	−18.62	−18.61	
	2	3	0.43	0.41	0.43	0.41	
		6	0.14	0.13	0.19	0.17	
	3	3	0.24	0.23	0.31	0.31	
		6	0.10	0.21	0.28	0.39	
High transportation cost	1	6	7.14	6.01	7.55	6.42	
	2	6	0.57	1.50	1.11	2.05	
	3	3	9.43	1.28	11.25	2.97	
		6	5.64	1.32	5.42	1.12	
	6	0.99	1.32	1.03	1.36		
Low inventory cost	2	6	2.16	2.06	2.41	2.31	
	3	3	5.06	3.42	5.14	3.51	
		6	2.81	2.81	3.68	3.68	
	Average		−0.14	−0.90	0.27	−0.50	

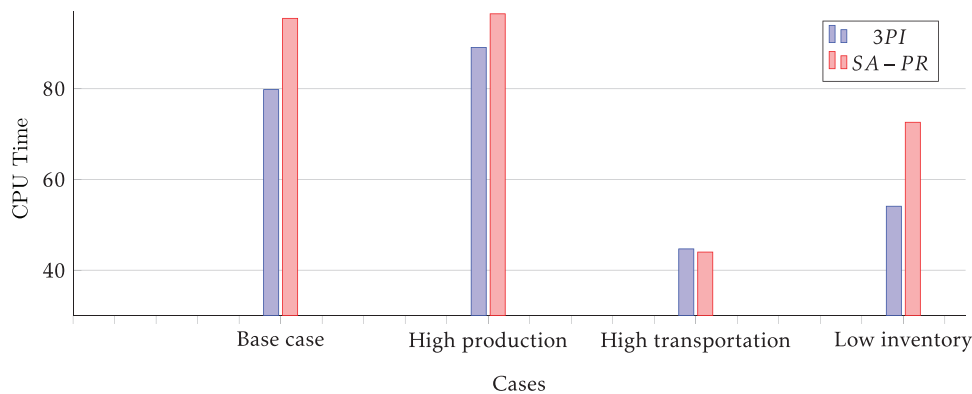


Fig. 5. Comparison of average CPU time between 3PI and SA-PR heuristics for all cases.

Figure 4 illustrates the average gap of the two heuristics, SA-PR and 3PI. Notably, SA-PR consistently outperforms 3PI across all cases, particularly in scenarios with high transportation costs. The smallest gap occurs in the high production cost case, where the number of produced products remains constant and the production cost is high.

In contrast, Fig. 5, which represents the average CPU time, reveals that SA-PR requires more time to find better solutions compared to 3PI, except in the high transportation cost case. Interestingly, the LKH-3 routing heuristic of Helsgaun (2017) stops prematurely, even if the resulting

Table 14

Performance comparison of proposed heuristics and CPLEX with 200 and 450 second time limits for large-sized instance

Cases	N_w	T	200 seconds		450 seconds	
			$\overline{\text{GAP}}^3$	GAP^s	$\overline{\text{GAP}}^3$	GAP^s
Base case	1	3	−30.61	−30.60	−29.46	−29.45
		6	−31.19	−30.61	−28.59	−27.99
	2	3	−0.80	−0.97	0.14	−0.02
		6	−1.80	−0.85	−0.68	0.27
	3	3	−1.10	−1.14	−0.40	−0.44
		6	−1.39	−1.06	−0.19	0.15
High production cost	1	3	−29.20	−29.20	−29.20	−29.20
		6	−29.10	−29.01	−29.05	−28.96
	2	3	−0.14	−0.17	0.00	−0.03
		6	−0.24	−0.07	−0.08	0.09
	3	3	−0.10	−0.10	−0.07	−0.07
		6	−0.45	−0.40	−0.14	−0.09
High transportation cost	1	3	3.00	2.62	3.48	3.10
		6	−5.67	−3.85	−2.59	−0.72
	2	3	0.02	1.79	0.18	1.95
		6	−1.78	−0.27	−2.19	−0.68
	3	3	−6.06	−7.93	−5.11	−6.99
		6	−3.46	−3.67	1.46	1.23
Low inventory cost	1	3	3.19	2.81	3.21	2.83
		6	2.16	2.72	2.17	2.72
	2	3	3.78	3.16	3.89	3.28
		6	1.28	2.36	1.69	2.78
	3	3	2.41	3.30	2.45	3.35
		6	1.66	1.84	1.90	2.08
Average			−5.23	−4.97	−4.47	−4.20

solution is far from optimal, indicating the trade-off between computational efficiency and solution quality.

These findings highlight the trade-offs between solution quality and computational efficiency for the two heuristics. While SA-PR achieves better solutions, it comes at the cost of increased computational time compared to 3PI.

5.5. Managerial insights

This section offers specific managerial insights concerning the distribution chain and the influence of the warehouse count, building upon the instances discussed in Section 5. To illustrate this, we have compared the current problem with two distinct real-world cases.

In the first case, each warehouse is assigned specific customers (from the second echelon), implying that each warehouse is responsible for delivering to a designated set of customers.

In the second case, both the manufacturing plant and the warehouses can deliver to all customers.

Below are the details of each case.

- The first case is named “nearest warehouse customers”: In several places, customers place their orders with the nearest warehouse, signifying that this particular warehouse is tasked with delivering to that specific customer. In this situation, we have linked each customer to the nearest warehouse, resulting in the creation of sets. To solve this problem, we have to change the warehouses’ sets from $N_{c'}$ to $N_{w,c'}$, where $N_{w,c'}$ represents the customers of each warehouse w in the mathematical formulation presented in Section 3.
- The second case is named “flexible dual delivery”: we have established a flexible distribution chain, allowing each customer (both the first and second echelons) to receive their orders through two options: either directly from the manufacturing plant with direct shipment or via one of the warehouses using vehicle routing. To solve this problem, we have made some changes to the model presented in Section 3 as follows:
 - Change both sets N_c and $N_{c'}$ to $N_T = N_c \cup N_{c'}$, here N_c , $N_{c'}$, and N_T represent the first-level customers, second-level customers, and the union of the sets of customers, respectively.
 - Replace constraints (3) and (5) with the following constraints to allow customers to receive their orders from the plant or warehouses:

$$I_{jt} = I_{j,t-1} + q_{jt} + \sum_{i \in N_w} \sum_{k \in K} r_{ijk t} - d_{jt} \quad \forall j \in N_T, \forall t \in T - \{1\}. \quad (47)$$

- Add a new decision variable u_{it} which is a binary variable that indicates if the manufacturing plant visited customer i in period t . The constraints are as follows:

$$v_{it} \leq Q_0 \times u_{it} \quad \forall i \in N_c \cup N_w, \forall t \in T. \quad (48)$$

- Modify constraints (11) to avoid split delivery as follows:

$$\sum_{i \in N_w} \sum_{k \in K} H_{ijk t} + u_{jt} \leq 1 \quad \forall j \in N_T, \forall t \in T. \quad (49)$$

5.5.1. Results

We utilized CPLEX to run the models for both scenarios: one with base case data and the other with high transportation costs data. Figures. 6–8 illustrate a comparison between the outcomes of two scenarios. These scenarios involve varying numbers of warehouses and consider both the base case and elevated transportation costs. Specifically, the comparison focuses on the total cost, transportation cost, and inventory cost. To demonstrate the impact of warehouse count, we established the baseline solution as the one-warehouse base case (in the case of high transportation cost, we set the high transportation case with one warehouse as the baseline solution), named as “basic case.” In each situation, we find the solution cost of using a specific number of warehouses and the Gap, which is calculated as follows:

$$Gap = 100 \times \frac{Sol_1^{Basic\ case} - Sol_{N_w}^{Situation}}{Sol_1^{Basic\ case}}. \quad (50)$$

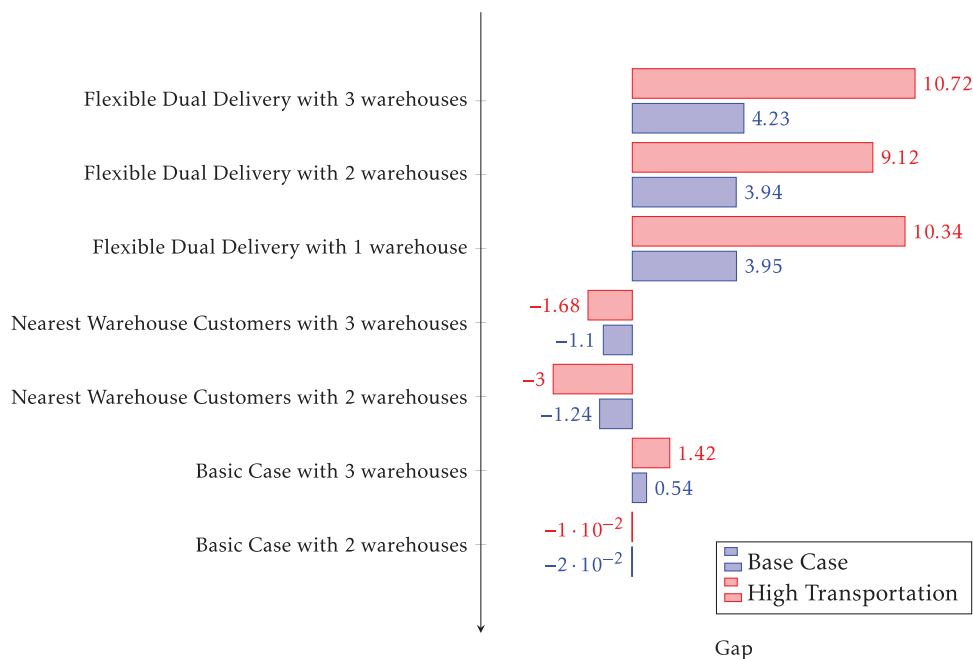


Fig. 6. Comparing total cost results of the three situations.

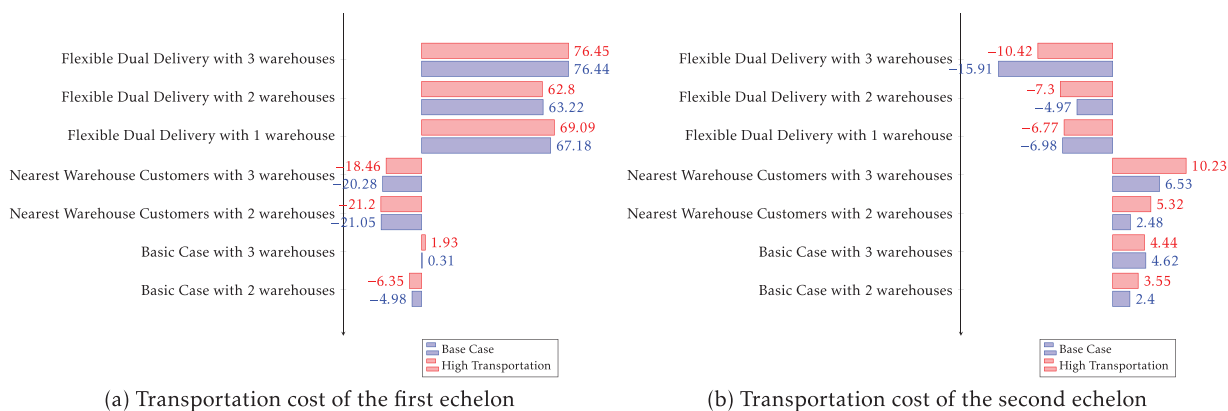


Fig. 7. Transportation cost results of the three situations.

When comparing the overall outcomes, it becomes evident that the “flexible dual delivery” situation outperforms the others in terms of cost reduction, achieving a decrease of 4.23% and 10.72% when having three warehouses, 3.94% and 9.12% with two warehouses, and 3.95% and 10.34% with one warehouse for the base and high transportation cost cases, respectively. This highlights the advantageous impact of distribution flexibility on the entire supply chain. This is due to the flexible dual delivery situation’s ability to utilize resources more effectively. It allows companies to better distribute their products and avoid shipping them long distances, which saves on transportation

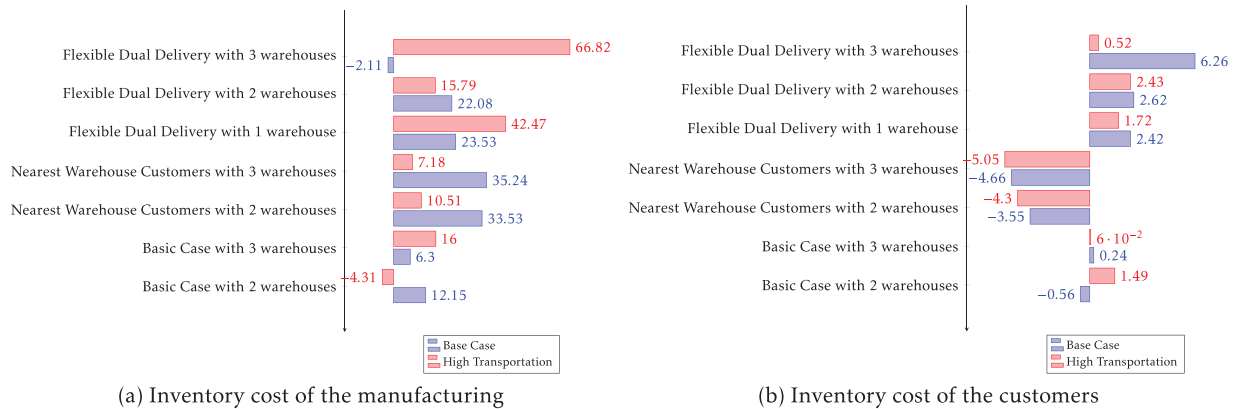


Fig. 8. Inventory cost results of the three situations.

costs until 76.45% for the first level with three warehouses. However, it increases the transportation cost of the second level with variation from 4.97% to 15.91%. Additionally, it allows companies to be more responsive to customer demand, which leads to lower inventory costs of 66.82%. While flexible dual delivery enhances resource utilization and responsiveness to customer demand, it also introduces operational complexity, inventory and logistical challenges, substantial costs, potential customer confusion, environmental impact, and security risks. On the other hand, the “nearest warehouse customers” situation is more expensive than the baseline solution, with a 1.24% cost increase in the base case and a 3% increase for the high transportation cost case. Although this approach achieves cost reductions in the transportation cost of the second level (up to 10.23%) and the inventory cost of the manufacturing plant (up to 35.24%), it simultaneously raises the costs of transportation at the first level (up to 21.20%) and customers’ inventory costs (up to 5.05%). These findings suggest that this scenario prioritizes making more deliveries in the first echelon to reduce manufacturing’s inventory costs while minimizing deliveries in the second echelon, leading to increased customers’ inventory costs.

These findings suggest that the basic situation strikes a balance between the “flexible dual delivery” and “nearest warehouse customers” approaches in terms of cost reduction and complexity.

Regarding the number of warehouses, it is evident within each situation that the three-warehouse configuration outperforms the two-warehouse configuration, and the two-warehouse configuration outperforms the one-warehouse configuration in terms of cost reduction, specifically in the base case. On the other hand, in the high transportation case, the situation with three warehouses is still the best.

5.6. Case study

This section presents a case study from Algeria, focusing on a soft drink manufacturing plant located in the country’s western region. This manufacturer produces soft drinks of different flavors (pineapple, strawberry, bitter, blackcurrant, orange, lemon, red apple, and green apple) and bottle types (33 CL, 1 L, and 2 L) in two main stages: liquid flavor preparation (stage I) and liquid bottling (stage II).

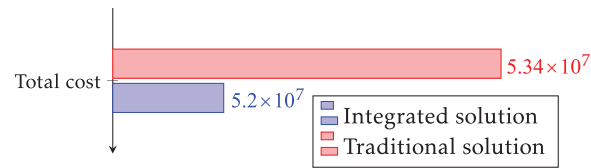


Fig. 9. Comparing total costs.

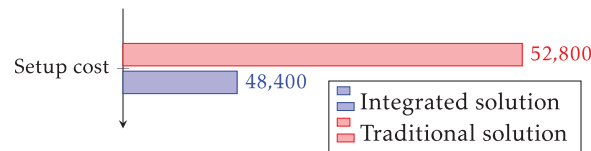


Fig. 10. Comparing setup costs.

In stage I, tanks of varying capacities prepare the liquid flavor (concentrate or syrup plus water). Technically, we cannot prepare two different liquid flavors in the same tank at the same time and the tank must be empty before preparing a new batch of liquid flavor.

Stage II involves bottling the liquid flavors in the filling line. Initially, the filling line is adjusted to produce soft drinks of a specific flavor in a specific bottle size. At each changeover of liquid flavor in the tanks and/or soft drink in the filling line, a sequence-dependent setup time (cleaning and/or machine adjustments) is necessary.

To fit our study, we consider the production process to be a production line, and we have focused on only one flavor (pineapple) and one size (1 L). One bottle of soft drink costs 59 Algerian Dinar (DA). Its production line, with a setup cost of 50,000 DA, can produce 666 bottles per hour, with a daily operating schedule of eight hours over five days a week.

The manufacturer serves one primary customer and a warehouse that distributes products across three cities, catering to a total of 24 customers, including grocery stores and soft drink distributors of various brands. The distances between the manufacturer, the primary customer, and the warehouse are 3.3 and 1.5 km, respectively.

The manufacturer's storage capacity is 10,000 bottles, with an inventory holding cost assumed to be 10% of the bottle price per month. The warehouse has a larger storage capacity of 20,000 bottles, and its inventory holding cost is 6 DA per bottle per month. The manufacturing plant has a fleet of vehicles with a capacity of 480 bottles, while the warehouse operates a fleet capable of transporting 6300 bottles. Transportation costs 10 DA per bottle per kilometer. The Appendix section outlines the distances between the company, warehouse, and customers.

5.6.1. Integrated versus traditional supply chain

In this section, we compare our study with the traditional supply chain of the Algerian company by utilizing the company's data. We selected the summer period when demand is high.

Based on the company's data, Figs. 9–12 show the results obtained from our integrated solution compared to the current approach being adopted by the company. Our findings show that the integrated solution lowers the total cost by 2.73% on average. This reduction is primarily due to better coordination between production, inventory, and transportation decisions.

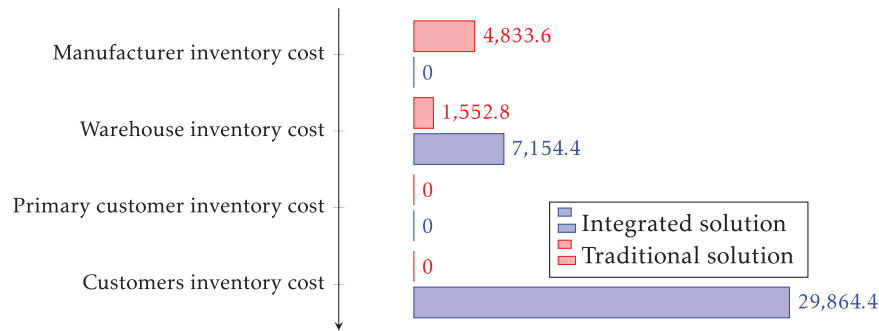


Fig. 11. Comparing inventory costs.

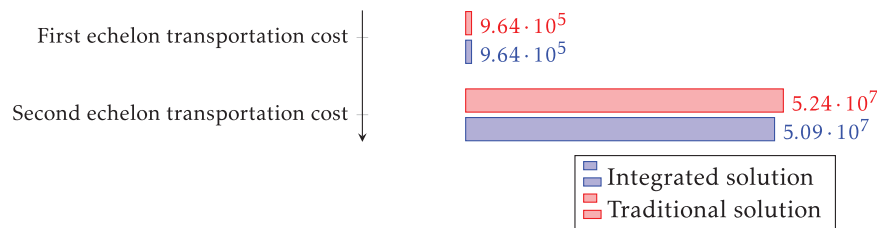


Fig. 12. Comparing transportation costs.

First, our method effectively reduces setup costs (see Fig. 10) by streamlining production scheduling and consolidating processes, resulting in fewer setup operations by reduction of one period. Second, transportation costs are reduced by 2.8%, achieved through fewer trips and shorter travel distances. The increase in total inventory holding costs shown in Fig. 11 (mainly at the customers) is largely compensated by the decrease in transportation and setup costs.

5.6.2. Evaluation of the company's plan

The company is considering the option of expanding by opening a second warehouse in another city. This new warehouse is expected to deliver six customers (distributors) in that city. In this section, we aim to determine whether this new warehouse is necessary or if the existing warehouse can handle the additional customers' demand. Additionally, we will assess whether the strategy of assigning each warehouse to its own set of customers is effective. We aim to identify the best solution for the company to adopt in order to minimize total costs.

Figures 13–16 present the results of three different strategies. The first strategy, called the “company's plan,” assigns each warehouse its own set of customers: the main warehouse serves customers in three cities, while the second warehouse serves customers in the new city. The second strategy, “the main warehouse serves customers,” involves using only the main warehouse to serve all customers in the four cities, effectively ignoring the new warehouse. The third strategy, “both warehouses serve customers,” allows both warehouses to serve all customers.

The total cost results show that the best option is not to use the new warehouse. When the model is given full flexibility in such a way that both warehouses can serve all customers, the best

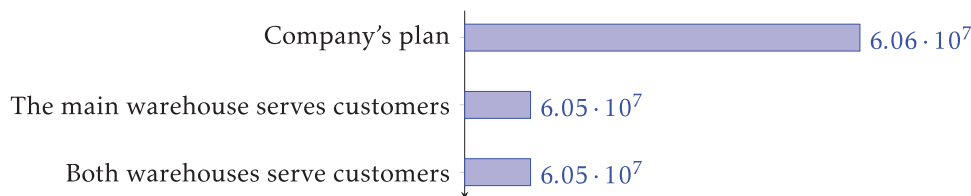


Fig. 13. Total cost results of the three strategies.

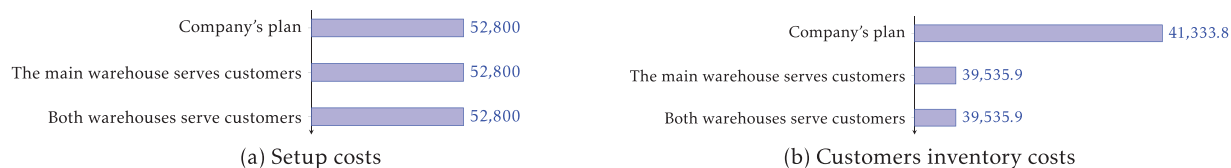


Fig. 14. setup cost and customers inventory cost results of the three strategies.

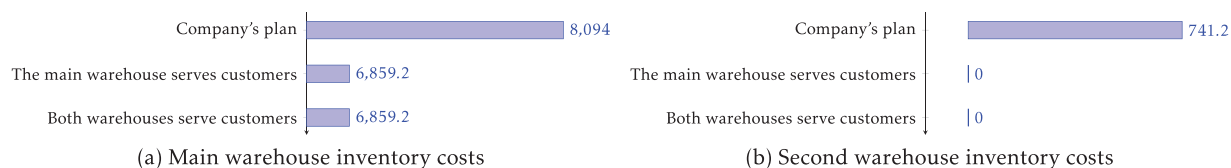


Fig. 15. Inventory cost results of the three strategies.

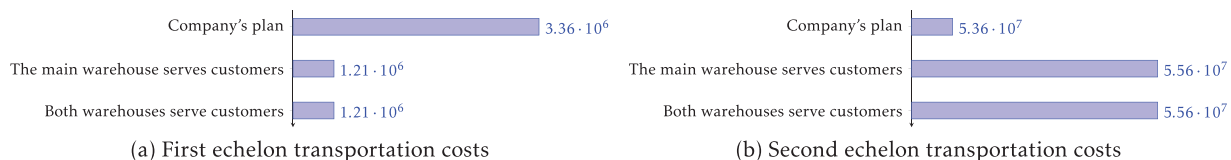


Fig. 16. Transportation cost results of the three strategies.

solution is to use only the company's current main warehouse. Hence, the company's plan is more expensive compared to using only the main warehouse. However, the gap between the costs of the two solutions is not very large: 0.2%. This led us to conclude that it might be interesting to open the second warehouse given other criteria than the total costs. Such criteria might include a better customer service level and better proximity to the market.

6. Conclusion

This research addressed the integration of production, inventory, and routing problems in a two-echelon supply chain with multiple delivery modes. The problem was effectively solved through an extended version of the two-phase iterative heuristic of Absi et al. (2015), and a metaheuristic inspired by Armentano et al. (2011), which combines SA-PR. Computational experiments have demonstrated that SA-PR achieves better solution quality than the extended heuristic while having

similar CPU time. Compared to CPLEX's solution, SA-PR found superior solutions in some cases and closely approximated CPLEX solutions in the majority of cases, with about 10 times less CPU time than CPLEX.

Managerial analysis has revealed that using the “flexible dual delivery” method can lead to significant cost reductions, from 3.94% to 10.72%. The cost reduction increases as the number of warehouses increases. This is due to the flexible dual delivery method's ability to utilize resources more effectively. It allows companies to better distribute their products and avoid shipping them long distances, saving transportation costs. Additionally, the findings suggest that the basic situation strikes a balance between the “flexible dual delivery” and “nearest warehouse customers” approaches in terms of cost reduction and complexity. We have presented a real-world case study involving an Algerian soft drink company, where we compared our proposed solution with the company's current approach. As a result, it was found that the integrated model can achieve a cost reduction of 2.73% on average.

The 2E-PRP can be expanded in various ways. One approach is to include uncertain demand and lead time, which allows for better management of scenarios where product demand and lead time are not precisely known (Adulyasak et al., 2015a). Another extension involves considering multi-echelon problems that deal with perishable products (Alvarez et al., 2022). Finally, the SA-PR algorithm could be improved to adapt to the two situations presented in Section 5.5 and to reduce CPU times and increase efficiency.

Acknowledgments

The authors would like to thank their corresponding institutions for supporting this research. They also thank the associate editor and reviewers for handling the manuscript and providing valuable feedback.

References

- Absi, N., Archetti, C., Dauzère-Pérès, S., Feillet, D., 2015. A two-phase iterative heuristic approach for the production routing problem. *Transportation Science* 49, 4, 784–795.
- Absi, N., Archetti, C., Dauzère-Pérès, S., Feillet, D., Speranza, M.G., 2018. Comparing sequential and integrated approaches for the production routing problem. *European Journal of Operational Research* 269, 2, 633–646.
- Adulyasak, Y., Cordeau, J.-F., Jans, R., 2014a. Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems. *INFORMS Journal on Computing* 26, 1, 103–120.
- Adulyasak, Y., Cordeau, J.-F., Jans, R., 2014b. Optimization-based adaptive large neighborhood search for the production routing problem. *Transportation Science* 48, 1, 20–45.
- Adulyasak, Y., Cordeau, J.-F., Jans, R., 2015a. Benders decomposition for production routing under demand uncertainty. *Operations Research* 63, 4, 851–867.
- Adulyasak, Y., Cordeau, J.-F., Jans, R., 2015b. The production routing problem: a review of formulations and solution algorithms. *Computers & Operations Research* 55, 141–152.
- Alvarez, A., Munari, P., Morabito, R., 2018. Iterated local search and simulated annealing algorithms for the inventory routing problem. *International Transactions in Operational Research* 25, 6, 1785–1809.
- Alvarez, A., Miranda, P., Rohmer, S.U., 2022. Production routing for perishable products. *Omega* 111, 102667.
- Ambrosino, D., Scutellà, M., 2005. Distribution network design: new problems and related models. *European Journal of Operational Research* 165, 3, 610–624.

- Archetti, C., Bertazzi, L., Paletta, G., Speranza, M.G., 2011. Analysis of the maximum level policy in a production-distribution system. *Computers & Operations Research* 38, 12, 1731–1746.
- Armentano, V.A., Shiguemoto, A.L., Løkketangen, A., 2011. Tabu search with path relinking for an integrated production–distribution problem. *Computers & Operations Research* 38, 8, 1199–1209.
- Bard, J.F., Nananukul, N., 2009. The integrated production–inventory–distribution–routing problem. *Journal of Scheduling* 12, 3, 257–280.
- Boudia, M., Prins, C., 2009. A memetic algorithm with dynamic population management for an integrated production–distribution problem. *European Journal of Operational Research* 195, 3, 703–715.
- Boudia, M., Louly, M.A.O., Prins, C., 2007. A reactive grasp and path relinking for a combined production–distribution problem. *Computers & Operations Research* 34, 11, 3402–3419.
- Boudia, M., Louly, M.A.O., Prins, C., 2008. Fast heuristics for a combined production planning and vehicle routing problem. *Production Planning and Control* 19, 2, 85–96.
- Bowersox, D.J., 1969. Physical distribution development, current status, potential. *Journal of Marketing* 33, 1, 63–70.
- Brahimi, N., Aouam, T., 2016. Multi-item production routing problem with backordering: a MILP approach. *International Journal of Production Research* 54, 4, 1076–1093.
- Brahimi, N., Absi, N., Dautère-Pérés, S., Nordli, A., 2017. Single-item dynamic lot-sizing problems: an updated survey. *European Journal of Operational Research* 263, 3, 838–863.
- Brekka, I., Randøy, S., Fagerholt, K., Thun, K., Vadseth, S.T., 2022. The fish feed production routing problem. *Computers & Operations Research* 144, 105806.
- Ceschia, S., Di Gaspero, L., Schaerf, A., 2017. Solving discrete lot-sizing and scheduling by simulated annealing and mixed integer programming. *Computers & Industrial Engineering* 114, 235–243.
- Ceschia, S., Di Gaspero, L., Rosati, R.M., Schaerf, A., 2026. Multi-neighborhood simulated annealing for the home healthcare routing and scheduling problem. *International Transactions in Operational Research* 33, 1, 38–67.
- Chandra, P., 1993. A dynamic distribution model with warehouse and customer replenishment requirements. *Journal of the Operational Research Society* 44, 7, 681–692.
- Chandra, P., Fisher, M.L., 1994. Coordination of production and distribution planning. *European Journal of Operational Research* 72, 3, 503–517.
- Chekoubi, Z., Trabelsi, W., Sauer, N., Majdoulina, I., 2022. The integrated production-inventory-routing problem with reverse logistics and remanufacturing: a two-phase decomposition heuristic. *Sustainability* 14, 20, 13563.
- Cuellar-Usaquén, D., Gomez, C., Álvarez-Martínez, D., 2023. A grasp/path-relinking algorithm for the traveling purchaser problem. *International Transactions in Operational Research* 30, 2, 831–857.
- Dantzig, G.B., Ramser, J.H., 1959. The truck dispatching problem. *Management Science* 6, 1, 80–91.
- Delahaye, D., Chaimatnan, S., Mongeau, M., 2019. Simulated annealing: from basics to applications. In Gendreau, M., Potvin, J.Y. (eds) *Handbook of Metaheuristics*. International Series in Operations Research & Management Science, vol 272. Springer, Cham, pp. 1–35.
- Fumero, F., Vercellis, C., 1999. Synchronized development of production, inventory, distribution schedules. *Transportation Science* 33, 3, 330–340.
- Glover, F., 1997. Tabu search and adaptive memory programming—advances, applications and challenges. *Interfaces in Computer Science and Operations Research: Advances in Metaheuristics, Optimization, and Stochastic Modeling Technologies*. Operations Research/Computer Science Interfaces Series, vol 7. Springer, Boston, MA, pp. 1–75.
- Gündüz Mengübaşı, G., Sörensen, K., Kotan, M., 2025. Conflict-free tow train routing in just-in-time assembly lines. *International Transactions in Operational Research* 32, 6, 3667–3692.
- Helsgaun, K., 2017. An extension of the Lin-Kernighan-Helsgaun TSP solver for constrained traveling salesman and vehicle routing problems. Roskilde University, Roskilde, Denmark, 12.
- Ho, S.C., Gendreau, M., 2006. Path relinking for the vehicle routing problem. *Journal of Heuristics* 12, 55–72.
- Hrabec, D., Hvattum, L.M., Hoff, A., 2022. The value of integrated planning for production, inventory, and routing decisions: a systematic review and meta-analysis. *International Journal of Production Economics* 248, 108468.
- Interian, R., Ribeiro, C.C., 2017. A grasp heuristic using path-relinking and restarts for the Steiner traveling salesman problem. *International Transactions in Operational Research* 24, 6, 1307–1323.

- Khan, M., Jaber, M.Y., Ahmad, A.-R., 2014. An integrated supply chain model with errors in quality inspection and learning in production. *Omega* 42, 1, 16–24.
- Kirkpatrick, S., 1984. Optimization by simulated annealing: quantitative studies. *Journal of Statistical Physics* 34, 975–986.
- Koulamas, C., Antony, S., Jaen, R., 1994. A survey of simulated annealing applications to operations research problems. *Omega* 22, 1, 41–56.
- Laporte, G., 2009. Fifty years of vehicle routing. *Transportation Science* 43, 4, 408–416.
- Lei, L., Liu, S., Ruszczyński, A., Park, S., 2006. On the integrated production, inventory, and distribution routing problem. *IIE Transactions* 38, 11, 955–970.
- Li, Y., Chu, F., Côté, J.-F., Coelho, L.C., Chu, C., 2020. The multi-plant perishable food production routing with packaging consideration. *International Journal of Production Economics* 221, 107472.
- Máximo, V.R., Nascimento, M.C., 2021. A hybrid adaptive iterated local search with diversification control to the capacitated vehicle routing problem. *European Journal of Operational Research* 294, 3, 1108–1119.
- Mor, A., Speranza, M.G., 2022. Vehicle routing problems over time: a survey. *Annals of Operations Research* 314, 1, 255–275.
- Neves-Moreira, F., Almada-Lobo, B., Cordeau, J.-F., Guimarães, L., Jans, R., 2019. Solving a large multi-product production-routing problem with delivery time windows. *Omega* 86, 154–172.
- Ozsen, L., Intrevado, P., Liu, S., 2020. Heuristics for a joint capacitated production, inventory, and distribution model with production setup times. *Journal of Supply Chain and Operations Management* 18, 1, 30.
- Pandey, D., Kulkarni, M.S., Vrat, P., 2011. A methodology for joint optimization for maintenance planning, process quality and production scheduling. *Computers & Industrial Engineering* 61, 4, 1098–1106.
- Peng, B., Lü, Z., Cheng, T.C.E., 2015. A tabu search/path relinking algorithm to solve the job shop scheduling problem. *Computers & Operations Research* 53, 154–164.
- Qiu, Y., Ni, M., Wang, L., Li, Q., Fang, X., Pardalos, P.M., 2018. Production routing problems with reverse logistics and remanufacturing. *Transportation Research Part E: Logistics and Transportation Review* 111, 87–100.
- Qiu, Y., Zhou, D., Du, Y., Liu, J., Pardalos, P.M., Qiao, J., 2021. The two-echelon production routing problem with cross-docking satellites. *Transportation Research Part E: Logistics and Transportation Review* 147, 102210.
- Ramos, B., Alves, C., Valerio de Carvalho, J., 2022. An arc flow formulation to the multitrip production, inventory, distribution, and routing problem with time windows. *International Transactions in Operational Research* 29, 1, 526–553.
- Ruokokoski, M., Solyali, O., Cordeau, J.-F., Jans, R., Süral, H., 2010. Efficient formulations and a branch-and-cut algorithm for a production-routing problem. Technical Report. Les Cahiers du GERAD G-2010-66, GERAD HEC Montréal, Montréal (Québec) Canada. [http://refhub.elsevier.com/S0925-5273\(22\)00061-5/sref76](http://refhub.elsevier.com/S0925-5273(22)00061-5/sref76)
- Schenekemberg, C.M., Scarpin, C.T., Pecora, J.E., Jr, Guimarães, T.A., Coelho, L.C., 2021. The two-echelon production-routing problem. *European Journal of Operational Research* 288, 2, 436–449.
- Senoussi, A., Mouss, N.K., Penz, B., Brahimi, N., Dauzère-Pérès, S., 2016. Modeling and solving a one-supplier multi-vehicle production-inventory-distribution problem with clustered retailers. *The International Journal of Advanced Manufacturing Technology* 85, 971–989.
- Shaabani, H., Kamalabadi, I.N., 2016. An efficient population-based simulated annealing algorithm for the multi-product multi-retailer perishable inventory routing problem. *Computers & Industrial Engineering* 99, 189–201.
- Shiguemoto, A.L., Armentano, V.A., 2010. A tabu search procedure for coordinating production, inventory and distribution routing problems. *International Transactions in Operational Research* 17, 2, 179–195.
- Wagner, H.M., Whitin, T.M., 1958. Dynamic version of the economic lot size model. *Management Science* 5, 1, 89–96.
- Zhou, Y., Xu, W., Fu, Z.-H., Zhou, M., 2022. Multi-neighborhood simulated annealing-based iterated local search for colored traveling salesman problems. *IEEE Transactions on Intelligent Transportation Systems* 23, 9, 16072–16082.

Appendix

Distances between warehouses and customers are as follows:

© 2025 The Author(s).

International Transactions in Operational Research published by John Wiley & Sons Ltd on behalf of International Federation of Operational Research Societies.

Nodes	W1	W2	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20	C21	C22	C23	C24	C25	C26	C27	C28	C29	C30
W1	0	81.9	6.2	7.8	8.5	9	10.2	6.5	6.3	8.4	172	170	168	166	166	166	166	168	165	167	167	169	77.7	73	70.2	72.4	85.8	83.9	79.7	84	79	79.4
W2	81.9	0	88.8	87.2	89.2	89.7	90.9	85.8	85.6	91.3	97.7	95.9	94.3	92.3	94.8	91.9	92.1	94	90.8	93.1	92.6	94.9	63.4	61.4	60.6	61.5	5.2	2.2	3.9	2.2	3.3	3.7
C1	6.2	88.8	0	2.2	3.9	4.4	5.6	6.7	6.7	6	177	176	174	172	174	171	172	174	170	173	172	174	72.1	74.3	71.4	73.1	91.4	89.5	85.3	89.6	84.6	85
C2	7.8	87.2	2.2	0	4.6	5.2	5.8	7.5	7.5	6.8	176	175	173	171	174	171	171	173	170	172	171	174	71.3	73.5	70.7	72.3	90.6	88.8	84.6	86.9	83.8	84.2
C3	8.5	89.2	3.9	4.6	0	0.55	3.2	3.4	3.4	2.7	179	178	176	174	176	173	174	175	172	175	174	176	74	76.2	73.3	75	93.3	91.4	87.2	91.5	86.5	86.9
C4	9	89.7	4.4	5.2	0.55	0	3.1	3.8	3.8	2.2	180	178	176	174	177	174	174	176	173	175	175	177	74.6	76.8	73.9	75.6	93.9	92	87.8	92.1	87.1	87.5
C5	10.2	90.9	5.6	5.8	3.2	3.1	0	6.1	6.7	5	182	180	178	176	179	176	176	178	175	177	176	179	76.4	78.6	75.7	77.4	95.7	93.8	89.6	93.9	88.9	89.3
C6	6.5	85.8	6.7	7.5	3.4	3.8	6.1	0	0.65	1.5	175	173	171	169	172	169	169	171	168	170	170	172	76.9	79.1	76.2	77.9	96.2	94.3	90.1	94.4	89.4	89.8
C7	6.3	85.6	6.7	7.5	3.4	3.8	6.7	0.65	0	2.1	182	181	178	177	179	176	176	178	175	177	177	179	76.9	79.1	76.3	77.9	96.1	94.2	90	94.3	89.3	89.7
C8	8.4	91.3	6	6.8	2.7	2.2	5	1.5	2.1	0	181	180	178	176	178	176	176	178	174	177	176	179	76.2	78.4	75.6	77.2	95.5	93.6	95.5	93.7	94.8	89.2
C9	172	97.7	177	176	179	180	182	175	182	181	0	2.1	4	8.3	11.3	9.7	8.6	10.6	8	5	16.1	13.5	86.8	86.7	86.8	85.5	92.6	99.7	102	99.8	101	101
C10	170	95.9	176	175	178	178	180	173	181	180	2.1	0	2.4	4.5	11.8	5.9	7	9.1	6.4	3.4	10.8	12	88.5	88.4	88.5	87.2	90.5	97.6	99.5	97.6	98.7	99.1
C11	168	94.3	174	173	176	176	178	171	178	178	4	2.4	0	3.1	8.8	4.5	6.1	8.2	5	2	9.4	11.4	87.5	87.5	87.6	86.3	89	96.1	98	96.2	97.3	97.7
C12	166	92.3	172	171	174	174	176	169	177	176	8.3	4.5	3.1	0	7.9	2.1	4.9	7	2.6	1.8	7	8.1	81	81	81.1	79.8	86.6	93.7	95.6	93.8	94.9	95.3
C13	169	94.8	174	174	176	177	179	172	179	178	11.3	11.8	8.8	7.9	0	8	3.3	1.7	4.7	9.1	3.4	0.35	72.7	72.6	72.7	71.4	88.8	95.9	97.8	96	97.1	97.5
C14	166	91.9	171	171	173	174	176	169	176	176	9.7	5.9	4.5	2.1	8	0	2.8	5.3	1.1	3	6.4	7.5	80.4	80.4	80.5	79.2	86	93.1	95	93.2	94.3	94.7
C15	166	92.1	172	171	174	174	176	169	176	176	8.6	7	6.1	4.9	3.3	2.8	0	2.3	2.4	6.1	5.3	3.3	80.5	80.5	80.6	79.3	86.1	93.2	95.1	93.3	94.4	94.8
C16	168	94	174	173	175	176	178	171	178	178	10.6	9.1	8.2	7	1.7	5.3	2.3	0	4.7	8.1	4.1	2.1	72.3	72.3	72.4	71.1	88.2	95.3	97.2	95.3	96.4	96.8
C17	165	90.8	170	170	172	173	175	168	175	174	8	6.4	5	2.6	4.7	1.1	2.4	4.7	0	3.5	5.6	4.8	79.6	79.6	79.7	78.4	85.2	92.3	94.2	92.4	93.5	93.9
C18	167	93.1	173	172	175	175	177	170	177	177	5	3.4	2	1.8	9.1	3	6.1	8.1	3.5	0	8.4	9.5	82.5	82.4	82.5	81.2	88	95.1	97	95.2	96.3	96.7
C19	167	92.6	172	171	174	175	176	170	177	177	16.1	10.8	9.4	7	3.4	6.4	5.3	4.1	5.6	8.4	0	3.7	72.4	72.4	72.5	71.2	86	93.1	95	93.2	94.3	94.7
C20	169	94.6	174	174	176	177	179	172	179	179	13.5	12	11.4	8.1	0.35	7.5	3.3	2.1	4.8	9.5	3.7	0	72.5	72.5	72.6	71.3	89.1	96.2	98.1	96.3	97.4	97.8
C21	77.7	63.4	72.1	71.3	74	74.6	76.4	76.9	76.9	76.2	86.8	88.5	87.5	81	72.7	80.4	80.5	72.3	79.6	82.5	72.4	72.5	0	4.2	2.8	2.2	66.9	69.1	69.5	69.9	69.9	69.9
C22	73	70.2	60.6	71.4	70.7	73.3	75.7	76.2	76.3	75.6	86.8	88.5	87.6	81.1	72.7	80.5	80.6	72.4	79.7	82.5	72.5	72.6	4.2	0	3	2.3	68.3	69.5	64.9	64.6	64.6	64.6
C23	72	70.2	60.6	71.4	70.7	73.3	75.7	76.2	76.3	75.6	86.8	88.5	87.6	81.1	72.7	80.5	80.6	72.4	79.7	82.5	72.5	72.6	2.8	3	0	2.1	64.5	62.6	61	60.7	60.3	60.7
C24	72.4	72.4	72.3	75	75.6	77.4	77.9	77.2	85.5	87.2	85.5	87.2	86.3	79.8	71.4	79.2	79.3	71.1	78.4	81.2	71.2	71.3	2.2	2.3	2.1	0	66.9	62.5	62.9	62.4	61.9	62.4
C25	58	58	2.2	68.5	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8
C26	83.9	2.2	68.5	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8	88.8
C27	79.7	3.9	85.3	84.6	87.2	87.8	89.6	90.1	90	95.5	102	99.5	98	95.6	97.8	95	95.1	97.2	94.2	97	95	98.1	63.5	64.9	61	62.7	7.5	2.8	0	2.7	1.1	0.072
C28	84	2.2	89.6	86.9	91.5	92.1	93.9	94.4	94.3	93.7	99.8	97.6	96.2	93.8	96	93.2	92.4	95.2	93.2	92.4	95.2	96.3	63.5	64.9	61	62.7	5.2	0.19	2.7	0	2.8	2.5
C29	79	3.3	84.6	83.8	86.5	87.1	88.9	89.4	89.3	94.8	101	98.7	97.3	94.9	97.1	94.3	94.4	96.4	93.5	96.3	94.3	97.4	62.8	64.1	60.3	61.9	8.6	2.9	1.1	2.8	0	1
C30	79.4	3.7	85	84.2	86.9	87.5	89.3	89.8	89.7	89.2	101	99.1	97.7	95.3	97.5	94.7	94.8	96.8	93.9	96.7	94.7	97.8	63.2	64.6	60.7	62.4	7.7	2.6	0.072	2.5	1	0