RAIRO-Oper. Res. 57 (2023) 1957–1981
https://doi.org/10.1051/ro/2023083

RAIRO Operations Research
www.rairo-ro.org

A ROBUST OPTIMIZATION APPROACH FOR THE PRODUCTION-ROUTING PROBLEM WITH LATERAL TRANSSHIPMENT AND OUTSOURCING *

Pedram Farghadani-Chaharsooghi[1,2,3,*] and Behrooz Karimi[3]

Abstract. Despite the fact that there is a large body of literature on the Production Routing Problem (PRP), we were struck by the dearth of research on outsource planning and lateral transshipment. This paper presents a mixed-integer linear programming model for incorporating outsourcing, lateral transshipment, back ordering, lost sales, and time windows into production routing problems. Then a robust optimization model will be introduced to overcome the detrimental effects of demand uncertainty. Considering the scale and complexity of the suggested problem, addressing it in a reasonable time was a challenge. Therefore, three matheuristic algorithms, including Genetic Algorithm (GA), Simulated Annealing (SA), and Modified Simulated Annealing (MSA), are developed for solving large-scale problems. Eventually, computational experiments on disparate instances are performed, and the results show the effectiveness and efficiency of the proposed algorithms. In other words, our recommended algorithms outperform the CPLEX solver in terms of the quality and time of obtaining the solutions.

Mathematics Subject Classification. 65K05, 90B05, 90B06, 90C90, 90C17.

Received July 21, 2022. Accepted June 7, 2023.

1. Introduction

Approximately six decades ago, Wagner and Whitin [1] introduced the "Lot Sizing Problem (LSP)", and Dantzig and Ramser [2] introduced the "Vehicle Routing Problem (VRP)" for the first time. The former refers to the determination of order quantities, which, depending on the type of problem, maybe the number of products that must be produced or purchased to meet Supply Chain (SC) requirements. The latter seeks to find suitable vehicle routes so that goods can be distributed to the customers at the lowest cost. The combination of these two issues, which includes deciding on the amount of production at the factory, replenishment decisions in each retailer and warehouse, and vehicle routing, is referred to as the "Production Routing Problem (PRP)" in the literature. In other words, this is the same "Inventory Routing Problem (IRP)" where the production cost, production setup time, and production decisions are not fixed [3]. It is worth noting that combining

Keywords. Supply chain Management, production-routing problem, outsourcing, lateral transshipment, demand uncertainty, robust optimization.

* *The Supplementary Material is available in electronic form at https://www.rairo-ro.org/10.1051/ro/2023083/olm.*

[1] Department of Mechanical, Industrial and Aerospace Engineering, Concordia University, Montreal, Quebec H3G 1M8, Canada.

[2] Interuniversity Research Center on Enterprise Networks, Logistics and Transportation (CIRRELT), Montreal, Canada.

[3] Department of Industrial Engineering and Management Systems, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran.

*Corresponding author: pedram.farghadani-chaharsooghi@mail.concordia.ca; pedramfarghadani@gmail.com

these two problems can significantly reduce supply chain costs. For instance, Kellogg's company, an American multinational food manufacturing company, as well as Frito-Lay, a subsidiary of PepsiCo, claimed to have saved several million dollars by applying PRP modelling in their supply chains [4]. Chandra [5] and Chandra and Fisher [6] investigated the benefits of combining these two problems.

The inventory system is a crucial component in any supply chain, and the strategies employed in this part of the chain can significantly affect the overall performance of the supply chain [7]. Typically, it is assumed that products can be transferred from one echelon to the next echelon in the supply chain. However, transshipment, also known as lateral transshipment in existing literature, refers to the movement of products within the same echelon of the supply chain. Transshipment usually occurs between an overstocked retailer and a retailer dealing with shortages, and it can help reduce inventory costs while improving service levels and customer satisfaction.

Herer *et al.* [8] divided lateral transshipment into two general categories:

(1) Emergency Lateral Transshipment (ELT): In this policy, the required goods will be transferred from another site in the same echelon as soon as facing a shortage in one location.
(2) Preventive Lateral Transshipment (PLT): This policy, unlike the previous one, is based on inventory forecasting so that if the retailer is likely to encounter a shortage, some goods will be sent from another retailer in the same echelon. This procedure minimizes the risk of exposure to scarcity. The article ahead utilizes the PLT policy in its model.

As competition intensifies in the supply chain industry, more attention is being given to outsourcing production and transportation. According to Coyle *et al.* [9], manufacturers and service providers in the United States see outsourcing as a viable option. There are various reasons why supply chain managers use outsourcing to achieve their cost and service level goals. Kaya [10] stated that some of these reasons include the quality of outsourced goods, the lack of technology needed to produce products, the inability to meet demand at the factory due to capacity constraints, and the lower cost of outsourcing compared to in-house production. In our problem, due to the factory's production capacity constraint, outsourcing is considered an additional tool to meet customer demand.

We present a novel approach by developing a robust optimization model that combines lot-sizing and vehicle routing problems in an uncertain environment. Our approach, inspired by the work of Bertsimas and Thiele [11], incorporates lateral transshipment between retailers with time windows and outsourcing as an alternative option to meet demand. This is the first time that such a comprehensive model has been developed. It is worth noting that outsourcing has been considered only for the manufacturing sector, and the routing decisions of the outsourced products are taken simultaneously with those of the manufactured products, which makes our model more efficient.

The paper is organized as follows. Section 2 presents the literature review, focusing on the PRPs and leading to the research gap that our paper fits. Problem description and mathematical formulations for both deterministic and stochastic models are given in Section 3. Section 4 provides detailed explanations about solution approaches used to solve the model, while Sections 5 and 6 contain numerical results and conclusions, respectively.

2. Literature review

The combination of production, inventory, delivery, and vehicle routing decisions are all covered by the production routing problem. In other words, the PRP is the IRP, where the amount of production will also be determined regarding its fixed and variable costs [4]. This subject was first proposed by Chandra [5]. Then, Chandra and Fisher [6] confirmed that the simultaneous combination of production, inventory and routing decisions would decrease between 3 and 20% of total supply chain costs. The combined optimization of lot-sizing and vehicle routing problems can provide advantages, but finding an exact solution is the challenge. As a result, many studies on IRPs and their extensions, such as PRPs, have concentrated on using heuristic and metaheuristic approaches.

Brahimi and Aouam [12] studied shortages in the form of back-ordering for the first time in PRPs. They referred to this as PRP-B and introduced a relax-and-fix heuristic method combined with a local search algorithm after developing two types of MIP formulations. One notable development in addressing global warming was the work of Qiu *et al.* [13], who addressed environmental concerns by applying the cap-and-trade law in PRPs. They also developed branch-and-price (B&P) heuristics to solve the model. Adulyasak *et al.* [4] utilized a novel algorithm, "optimization-based adaptive large neighbourhood search", to solve the PRP. The model's time horizon is a finite, discrete time period, and it assumes that demands are dynamic. Absi *et al.* [14] conducted a study comparing the performance of an integrated approach with two sequential approaches. One sequential approach optimized production decisions first, while the other optimized distribution decisions first. The authors used a state-of-the-art heuristic algorithm for the PRP to solve the integrated problem. Chitsaz *et al.* [15] proposed a decomposition matheuristic initially designed for the assembly routing problem, showing that it could also effectively solve the PRP. Recently, there has been an increasing focus on incorporating workforce planning into PRPs. This has become a popular area of research as it offers significant potential for improving production and delivery efficiency [16, 17]. Farghadani-Chaharsooghi *et al.* [17] combined workforce planning with the PRP to address the specific needs of a company that specializes in processing and delivering organic fruits and vegetables. A combination of the Progressive Hedging Algorithm (PHA), Benders Decomposition (BD), and GA is employed to solve the model. Brekkå *et al.* [18] conducted research on a particular issue faced by Norwegian salmon feed manufacturers. The problem they focused on was the need to identify the most cost-effective methods for creating and distributing fish feed products. The authors created an adaptive large neighborhood search algorithm to address this problem using a decomposition-based approach. Vadseth *et al.* [19] recently created a multi-start matheuristic to solve the PRP using two MILPs. One of them constructs solutions, and another one improves solutions.

Adulyasak *et al.* [3] stated that there are only a few exact quantitative methods available to solve small-sized PRPs. Adulyasak *et al.* [20] investigated the PRP, for the first time, by considering the order-up-to-level inventory strategy in both single-vehicle and multi-vehicle modes. In other words, they solved the PRP using the order-up-to-level (OU) inventory strategy and compared it to the same problem but with the maximum level (ML) inventory strategy that had been studied by that time. They employed a Branch-and-Cut (B&C) algorithm for both OU and ML strategies. Adulyasak *et al.* [3] incorporated uncertainty in demands and solved the model by a BD algorithm. Dayarian and Desaulniers [21] created a model for a catering service company that had real-world constraints such as time windows and multiple trips. They introduced an exact branch price and-cut algorithm to solve the problem in an efficient manner. Qiu *et al.* [22] tackled a multi-product multi-vehicle PRP that takes into account the cost of restarting machines when changing between product lines. They solved the problem using a B&C algorithm. Similarly, Darvish *et al.* [23] and Qiu *et al.* [24] utilized a B&C algorithm for solving PRPs with reverse logistics and sustainable PRPs, respectively. Schenekemberg *et al.* [25] devised a new exact algorithm for solving a two-echelon production routing problem. They achieved this by combining the traditional B&C algorithm with local search procedures. Most recently, Manousakis *et al.* [26] presented a new version of the PRP called the cyclic PRP (CPRP) with a two-commodity flow formulation. This problem is similar to the basic PRP as it involves generating repeatable cyclic production and delivery schedules. The researchers developed a solution method using a branch and cut algorithm to solve this problem. For those interested in PRP, we recommend referring to the in-depth reviews by Adulyasak *et al.* [3] and Hrabec *et al.* [27] in the field of PRP literature. These resources provide valuable and detailed insights into the topic.

Our review of prior research on PRPs reveals that no previous study has considered the options of transshipment and outsourcing. To address this gap and better reflect practical constraints, we propose a new model for the PRP that considers retailers with time windows. Furthermore, research has shown that even medium-sized VRPs cannot be solved exactly [28]. Given this, we developed three methods – GA, SA, and MSA – to tackle the PRP with transshipment between retailers. A summary of the PRP literature and the significance of the current paper is presented in Table 1.

Table 1. Literature review summary.

Article	Multi product	Lost sales	Back order	Heterogeneous fleet	Demand uncertainty	Transshipment	Time window	Outsourcing	Setup time	Solution approach
Brahimi and Aouam [12]	✓		✓							Relax-and-Fix Heuristic Algorithm
Qiu *et al.* [13]		✓								*Ad Hoc* Label-Setting Algorithm
Qiu *et al.* [22]	✓									B&C Algorithm
Qiu *et al.* [24]				✓						B&C Algorithm
Dayarian and Desaulniers [21]	✓								✓	Branch-Price-and-Cut Algorithm
Darvish *et al.* [23]									✓	B&C Algorithm
Schenekemberg *et al.* [25]	✓			✓						B&C Algorithm
Brekkå *et al.* [18]	✓	✓		✓			✓		✓	Adaptive Large Neighborhood Search
Farghadani-Chaharsooghi *et al.* [17]	✓	✓		✓	✓		✓			PHA + BD + GA
Majidi *et al.* [16]	✓			✓	✓		✓			Fuzzy Domination SelfLearning NSGA-II
Current Study	✓	✓	✓	✓	✓	✓	✓	✓	✓	GA / SA / MSA

3. Problem description and mathematical formulation

A production-distribution system is assumed, in which a manufacturing plant (represented by zero index) delivers its P different products, to n retailers, over discrete interval time series with T periods using K different vehicles. The current production routing problem will be defined on the complete graph $G = (N_0, E)$, where N_0 represents the set of n retailers and a production plant, and E is the set of all edges of the chain defined as follows: $E = \{(i, j) : i, j \in N_0, i \neq j\}$.

Problem assumptions

(1) Split delivery is not allowed.
(2) Each retailer has its own time window.
(3) The transportation fleet is heterogeneous.
(4) Each vehicle can only have one trip per period.
(5) Transshipment is possible between all retailers.
(6) The trip of each vehicle starts and ends at the plant.
(7) Each retailer must be visited by the vehicles at most once per period.
(8) The capacities of retailers' inventory and the plant are limited.
(9) Back-ordering and lost sales are permitted in all periods except the final one, where only lost sales are taken into account.
(10) Each product must be entirely produced in each period. In other words, having semi-finished products are not permitted in this model.
(11) Each product's production involves two distinct phases – setup and production time. These phases are unique to each product type and are not dependent on the order in which operations are performed.
(12) Suppose there is an inventory transfer from retailer A to retailer B for a particular product during a given period. In that case, transshipment from retailer B to A for that same product is not permitted during that period.

In this section, we first introduce indices, parameters, and decision variables used in the model. Then we present the mathematical model.

Sets

T	Set of time periods
I	Set of retailers
I_0	Set of all nodes, including retailers and the factory plant
K	Set of vehicles
P	Set of products
R	Set of available resources, *i.e.*, time, for production

Parameters

f_p	Fixed production setup cost of product p
b_p	Unit production cost of product p
Oc_p	Unit outsourcing cost of product p
h_{ip}	Unit inventory holding cost at node i for product p per period
π_p	Unit back-ordering cost of product p per period
π'_p	Unit shortage (lost sale) cost of product p per period
u_{ip}	Inventory capacity at node i for product p
d_{ipt}	Customer demand at retailer i for products p in period t
c_{ij}	Transportation cost over arc (i, j)
Ftc	Fixed transportation cost
Vtc	Variable transportation cost
Q_k	Capacity of vehicle k
$[e_{it}, l_{it}]$	Beginning and end of time window for the retailer i in period t
τ_{ijk}	Required time for vehicle k to service at node i and arrive at the retailer j
α	Percentage of shortages in each period, which will be compensated in the next period (Therefore $(1-\alpha)\%$ of the shortages will also be considered as lost sales)
Ra_{rt}	Amount of available resource (time) r in period t
Rc_{pr}	The consumption rate of each product p from resource (time) r
Rs_{pr}	Setup time of product p to use resource r
Fct	Fixed cost of transshipment
Vct	Variable cost of transshipment
M	A big enough number

Decision variables

P_{pt}	Production quantity of product p in period t
OP_{pt}	Outsourced quantity of product p in period t
q_{ipkt}	Delivery quantity of products p to retailer i in period t by vehicle k
I_{ipt}	Inventory quantity of product p at retailer i at the end of period t
I_{0pt}	Inventory quantity of product p at the factory at the end of period t
T_{ijpt}	The amount of product p which is transshipped from retailer i to retailer j in period t
LB_{ipt}	Shortage amount of product p at node i at the end of period t
w_{ikt}	Service start time of vehicle k at node i in period t
y_{pt}	1 if product p is produced in period t, 0 otherwise
x_{ijkt}	1 if vehicle k travels directly from node i to node j in period t, 0 otherwise
TR_{ijpt}	1 if product p in period t is transshipped from retailer i to the retailer j, 0 otherwise

z_{ikt} 1 if node i is visited by vehicle type k in period t, 0 otherwise

3.1. Deterministic Model (DM)

We formulted the production routing problem by considering time window, lateral transshipment, demand uncertainty, and outsourcing as follows.

$$\begin{aligned}\text{Minimizing } & \sum_{t=1}^{T}\sum_{p=1}^{P}\left(f_p \times y_{pt} + b_p \times P_{pt} + Oc_p \times \mathrm{OP}_{pt} + \sum_{i=0}^{n}(h_{ip} \times I_{ipt})\right)\\ & + \sum_{t=1}^{T}\left(\sum_{(i,j)\in E}\sum_{p=1}^{P} Fct \times \mathrm{TR}_{ijpt} + \sum_{(i,j)\in E}\sum_{p=1}^{P} Vct \times c_{ij} \times \mathrm{TR}_{ijpt}\right)\\ & + \sum_{t=1}^{T}\left(\sum_{k=1}^{K}\sum_{i=1}^{n} Ftc \times x_{0ikt} + \sum_{k=1}^{K}\sum_{(i,j)\in E} Vtc \times c_{ij} \times x_{ijkt}\right)\\ & + \sum_{t=1}^{T-1}\left(\sum_{i=1}^{n}\sum_{p=1}^{P}(1-\alpha)\pi'_p \times \mathrm{LB}_{ipt} + \alpha \times \pi_p \times \mathrm{LB}_{ipt}\right) + \sum_{i=1}^{n}\sum_{p=1}^{P}\left(\pi'_p \times \mathrm{LB}_{ipT}\right)\end{aligned} \tag{3.1}$$

s.t.

$$I_{0p(t-1)} + P_{pt} + \mathrm{OP}_{pt} = \sum_{i=1}^{n}\sum_{k=1}^{K} q_{ipkt} + I_{0pt}; \qquad \forall p \in P, t \in T \tag{3.2}$$

$$\begin{aligned}& I_{ip(t-1)} + \sum_{k=1}^{K} q_{ipkt} + \sum_{j=1}^{n} T_{jipt} - \sum_{j=1}^{n} T_{ijpt} - \alpha \times \mathrm{LB}_{ip(t-1)}\\ & \quad = d_{ipt} + I_{ipt} - \mathrm{LB}_{ipt}; \qquad \forall i \in I, p \in P, t \in T\end{aligned} \tag{3.3}$$

$$I_{ipt} \le u_{ip}; \qquad \forall i \in I_0, p \in P, t \in T \tag{3.4}$$

$$P_{pt} \le M \times y_{pt}; \qquad \forall p \in P, t \in T \tag{3.5}$$

$$\sum_{p=1}^{P}(Rs_{pr} \times y_{pt} + Rc_{pr} \times P_{pt}) \le Ra_{rt}; \qquad \forall r \in R, t \in T \tag{3.6}$$

$$\sum_{i=1}^{n}\sum_{p=1}^{P} q_{ipkt} \le Q_k; \qquad \forall t \in T, k \in K \tag{3.7}$$

$$q_{ipkt} \le M \times z_{ikt}; \qquad \forall i \in I, p \in P, t \in T, k \in K \tag{3.8}$$

$$\sum_{j=1}^{n} T_{ijpt} \le I_{ip(t-1)} + \sum_{k=1}^{K} q_{ipkt}; \qquad \forall i \in I, p \in P, t \in T; i \ne j \tag{3.9}$$

$$T_{jipt} \le M \times \mathrm{TR}_{jipt}; \qquad \forall \{i,j\} \in I, p \in P, t \in T; i \ne j \tag{3.10}$$

$$\sum_{\substack{j=0\\ j\ne i}}^{n} x_{ijkt} = z_{ikt}; \qquad \forall i \in I, t \in T, k \in K \tag{3.11}$$

$$\sum_{\substack{j=0\\ j\ne i}}^{n} x_{jikt} = \sum_{\substack{j=0\\ j\ne i}}^{n} x_{ijkt}; \qquad \forall i \in I, t \in T, k \in K \tag{3.12}$$

$$\sum_{\substack{j=1\\j\neq i}}^{n} x_{0jkt} \leq 1; \qquad \forall t \in T, k \in K \tag{3.13}$$

$$\sum_{k=1}^{K} z_{ikt} \leq 1; \qquad \forall i \in I, t \in T \tag{3.14}$$

$$e_{it} \times z_{ikt} \leq w_{ikt}; \qquad \forall i \in I, t \in T, k \in K \tag{3.15}$$

$$w_{ikt} \leq l_{it} \times z_{ikt}; \qquad \forall i \in I, t \in T, k \in K \tag{3.16}$$

$$w_{ikt} + \tau_{ijk} \leq w_{jkt} + M \times (1 - x_{ijkt}); \qquad \forall \{i,j\} \in I, t \in T, k \in K \tag{3.17}$$

$$q_{ipkt}, P_{pt}, I_{ipt}, T_{ijpt}, \mathrm{LB}_{ipt}, \mathrm{OP}_{pt}, w_{ikt} \geq 0; \qquad \forall \{i,j\} \in I, p \in P, t \in T, k \in K \tag{3.18}$$

$$x_{ijkt}, z_{ikt}, y_{pt} \in \{0,1\}; \qquad \forall i \in I_0, p \in P, t \in T, k \in K. \tag{3.19}$$

The objective function (3.1) minimizes the total costs of production, maintenance, outsourcing, shortage, transshipment, and routing. Constraints (3.2) and (3.3) guarantee the inventory balance at the manufacturing plant and each retailer, respectively. Constraint (3.4) consider the inventory capacity of the manufacturing plant and each retailer. Constraint (3.5) create a link between production and its fixed cost. The quantity of production of each product in the factory is also controlled by the constraint (3.6) regarding available and required resources, *i.e.*, time, to produce each product in each period. The capacity of the vehicles is respected by constraint (3.7). Constraint (3.8) state that a positive delivery to a retailer is permitted if the retailer is visited during that period. Constraint (3.9) indicate that the amount of product which can be transshipped from retailer A to retailer B should not be higher than retailer A's inventory in the previous period. Constraint (3.10) calculate the fixed cost resulting from inventory transshipment between retailers. The constraints (3.11) and (3.12) ensure that each retailer is visited only once during each period if necessary, and once a vehicle enters a retailer, it must leave it. Constraint (3.13) enforce each vehicle to leave the plant only once during each period. Each retailer can only be visited by one vehicle at each period, which applies to the model by constraint (3.14). Each vehicle is only allowed to meet and serve retailers within the specified time window, subject to restrictions (3.15)–(3.17). Constraint (3.17) do also act as a sub-tour elimination constraint, and therefore. Constraints (3.18) and (3.19) are for nonnegativity of variables and binary restrictions, respectively.

3.2. A robust approach for tackling demand uncertainty in the Stochastic Model (SM)

Initial steps in the field of robust optimization were taken by Soyster [29], but the term "robust optimization" was first proposed by Mulvey *et al.* [30]. In the approach presented by Soyster [29] non of the constraints will be violated. On the other hand, such an approach to optimization problems results in a highly conservative solution. Therefore, there have been several attempts to reduce the conservatism degree of the objective function in Soyster's approach [31, 32]. One of the disadvantages of the approaches mentioned before is their poor computational traceability due to the existence of nonlinear models, especially for discrete problems. Thus, to deal with the conservative approach as well as the difficulties of nonlinear models presented in previous works, Bertsimas and Sim (henceforth Bertsimas and Sim's approach) presented a linear robust optimization approach based on polyhedral uncertainty sets [33, 34]. It should be noted that the method of Bertsimas and Sim is also named "budget of uncertainty". This kind of robust optimization is also called "interval robust optimization". Because in this type of optimization, the uncertain variables will be set at specific intervals. Bertsimas and Sim [34] claimed that the probability that all uncertain parameters would change simultaneously and fall into their worst value is very low and thus defined a parameter called budget of uncertainty, *i.e.*, Γ. Based on the parameter's definition, the sum of the variations of the uncertain parameters cannot be greater than Γ. Γ is also used to control the degree of conservatism in the model. This parameter can be set between zero and the number of uncertain parameters in the constraints and does not necessarily have to be an integer. Therefore, the maximum number of parameters that will be allowed to get their worst value is determined by Γ. In other words, the larger this parameter is, the lower the probability of constraint violation will be [35].

It is now necessary to implement the Bertsimas and Sim's approach, on the original model, in which the constraint (3.3) will first has to be rewritten as follows.

$$I_{ip(t-1)} + \sum_{k=1}^{K} q_{ipkt} + \sum_{j=1}^{n} T_{jipt} - \sum_{j=1}^{n} T_{ijpt} - \alpha \times \mathrm{LB}_{ip(t-1)} - I_ipt + \mathrm{LB}_{ipt} = d_{ipt}; \qquad \forall i \in I, p \in P, t \in T. \quad (3.20)$$

Since the main objective function is minimization, the model will not produce more products than demands, so the equality constraint (3.20) can be represented as inequality by considering an additional variable with a constant value of one.

$$-\left(I_{ip(t-1)} + \sum_{k=1}^{K} q_{ipkt} + \sum_{j=1}^{n} T_{jipt} - \sum_{j=1}^{n} T_{ijpt} - \alpha \times \mathrm{LB}_{ip(t-1)} - I_ipt + \mathrm{LB}_{ipt}\right) + \tilde{d}_{ipt} \times 1 \leq 0; \qquad \forall i \in I, p \in P, t \in T. \quad (3.21)$$

The constraint (3.21) has only one uncertain coefficient $\tilde{d}_{ipt}$. Therefore, the budget of uncertainty Γ_{ipt} will take a value in the range of [0,1]. We also assumed that demands could take value in the range of $[d_{ipt} - \hat{d}_{ipt}, d_{ipt} + \hat{d}_{ipt}]$ where d_{ipt} and $\hat{d}_{ipt}$ are the nominal value and the variation of demands, respectively. However, it has been proven that this approach is not efficient enough to deal with demand uncertainty [36]. To solve this problem, the approach proposed by Bertsimas and Thiele [11] will be implemented in the mathematical model instead of the previous one. For this purpose, we first define a free variable called net inventory ($\mathrm{NI}_{ipt} = I_{ipt} - \mathrm{LB}_{ipt}$) as follows.

$$\mathrm{NI}_{ipt} = \sum_{\theta=1}^{t}\sum_{k=1}^{K} q_{ipk\theta} + \sum_{\theta=1}^{t}\sum_{j=1}^{n} T_{jip\theta} - \sum_{\theta=1}^{t}\sum_{j=1}^{n} T_{ijp\theta} - \sum_{\theta=1}^{t} d_{ip\theta}; \qquad \forall i \in I, p \in P, t \in T. \quad (3.22)$$

Finally, the amount of inventory will be obtained through constraints (3.23) and (3.24).

$$\sum_{\theta=1}^{t}\sum_{k=1}^{K} q_{ipk\theta} + \sum_{\theta=1}^{t}\sum_{j=1}^{n} T_{jip\theta} - \sum_{\theta=1}^{t}\sum_{j=1}^{n} T_{ijp\theta} - \sum_{\theta=1}^{t} d_{ip\theta} \leq u_{ip}; \qquad \forall i \in I, p \in P, t \in T \quad (3.23)$$

$$\sum_{j=1}^{n} T_{ijpt} \leq \left(\sum_{\theta=1}^{t-1}\sum_{k=1}^{K} q_{ipk\theta} + \sum_{\theta=1}^{t-1}\sum_{j=1}^{n} T_{jip\theta} - \sum_{\theta=1}^{t-1}\sum_{j=1}^{n} T_{ijp\theta} - \sum_{\theta=1}^{t-1} d_{ip\theta}\right) + \sum_{k=1}^{K} q_{ipkt}; \qquad \forall i \in I, p \in P, t \in T; i \neq j. \quad (3.24)$$

Now having net inventory and defining auxiliary variable R_{ipt} as:

$$R_{ipt} = \max\big((h_{ip} \times I_{ipt}), (\alpha \times -\pi_p \times \mathrm{LB}_{ipt} + (1-\alpha) \times -\pi'_p \times \mathrm{LB}_{ipt})\big).$$

It is possible to calculate the inventory and shortage cost from the equations (3.25)–(3.27).

$$R_{ipt} \geq h_{ip} \times I_{ipt} = h_{ip} \times \left(\sum_{\theta=1}^{t}\sum_{k=1}^{K} q_{ipk\theta} + \sum_{\theta=1}^{t}\sum_{j=1}^{n} T_{jip\theta} - \sum_{\theta=1}^{t}\sum_{j=1}^{n} T_{ijp\theta} - \sum_{\theta=1}^{t} d_{ip\theta}\right); \qquad \forall i \in I, p \in P, t \in T \quad (3.25)$$

$$R_{ipt} \geq \alpha \times -\pi_p \times \mathrm{LB}_{ipt} + (1-\alpha) \times -\pi'_p \times \mathrm{LB}_{ipt}$$

$$
\begin{aligned}
&= \alpha \times -\pi_p \times \left(\sum_{\theta=1}^{t} \sum_{k=1}^{K} q_{ipk\theta} + \sum_{\theta=1}^{t} \sum_{j=1}^{n} T_{jip\theta} - \sum_{\theta=1}^{t} \sum_{j=1}^{n} T_{ijp\theta} - \sum_{\theta=1}^{t} d_{ip\theta} \right) \\
&\quad + (1-\alpha) \times -\pi'_p \\
&\quad \times \left(\sum_{\theta=1}^{t} \sum_{k=1}^{K} q_{ipk\theta} + \sum_{\theta=1}^{t} \sum_{j=1}^{n} T_{jip\theta} - \sum_{\theta=1}^{t} \sum_{j=1}^{n} T_{ijp\theta} - \sum_{\theta=1}^{t} d_{ip\theta} \right); \qquad \forall i \in I, p \in P, t \in T; t \neq T \quad (3.26)
\end{aligned}
$$

$$
\begin{aligned}
R_{ipt} &\geq -\pi'_p \times \mathrm{LB}_{ipt} = -\pi'_p \\
&\quad \times \left(\sum_{\theta=1}^{t} \sum_{k=1}^{K} q_{ipk\theta} + \sum_{\theta=1}^{t} \sum_{j=1}^{n} T_{jip\theta} - \sum_{\theta=1}^{t} \sum_{j=1}^{n} T_{ijp\theta} - \sum_{\theta=1}^{t} d_{ip\theta} \right); \qquad \forall i \in I, p \in P, t = T. \quad (3.27)
\end{aligned}
$$

Since there is $\sum_{\theta=1}^{t} d_{ip\theta}$ in the constraints (3.25)–(3.27), it means instead of having one uncertain variable in each constraint, we will have t uncertain variables. Therefore, the budget of uncertainty could take value in the range of $[0, t]$ [37]. Based on the model presented by Bertsimas and Thiele [11], we assume that the budget of uncertainty is as follows.

(1) $\Gamma_{ip1} \leq \Gamma_{ip2} \leq \cdots \leq \Gamma_{ipT}$; It means that the budget of uncertainty will not decrease over the time horizon.
(2) $\Gamma_{ipt} - \Gamma_{ip(t-1)} \leq 1$; It indicates that the increase in the budget of uncertainty is less than the increase in the number of periods.

Finally, the mathematical model of the PRP by considering time window, outsourcing, lateral transshipment, demand uncertainty, and the initial inventory of $Inv0$ is as follows:

$$
\begin{aligned}
\text{Minimizing } & \sum_{t=1}^{T} \sum_{p=1}^{P} (f_p \times y_{pt} + b_p \times P_{pt} + Oc_p \times \mathrm{OP}_{pt} + h_{0p} \times I_{0pt}) \\
& + \sum_{t=1}^{T} \left(\sum_{(i,j)\in E} \sum_{p=1}^{P} Fct \times \mathrm{TR}_{ijpt} + \sum_{(i,j)\in E} \sum_{p=1}^{P} Vct \times c_{ij} \times \mathrm{TR}_{ijpt} \right) \\
& + \sum_{t=1}^{T} \left(\sum_{k=1}^{K} \sum_{i=1}^{n} Ftc \times x_{0ikt} + \sum_{k=1}^{K} \sum_{(i,j)\in E} Vtc \times c_{ij} \times x_{ijkt} \right) + \sum_{t=1}^{T} \sum_{i=1}^{n} \sum_{p=1}^{P} R_{ipt} \qquad (3.28)
\end{aligned}
$$

s.t.

$$
\begin{aligned}
R_{ipt} &\geq h_{ip} \times \Bigg(Inv0 + \sum_{\theta=1}^{t} \sum_{k=1}^{K} q_{ipk\theta} + \sum_{\theta=1}^{t} \sum_{j=1}^{n} T_{jip\theta} \\
&\quad - \sum_{\theta=1}^{t} \sum_{j=1}^{n} T_{ijp\theta} - \sum_{\theta=1}^{t} d_{ip\theta} + (\Gamma_{ipt} \times v_{ipt}) + \sum_{\theta=1}^{t} g_{ip\theta t} \Bigg); \qquad \forall i \in I, p \in P, t \in T \qquad (3.29)
\end{aligned}
$$

$$
\begin{aligned}
R_{ipt} &\geq \alpha \times -\pi_p \times \Bigg(Inv0 + \sum_{\theta=1}^{t} \sum_{k=1}^{K} q_{ipk\theta} + \sum_{\theta=1}^{t} \sum_{j=1}^{n} T_{jip\theta} \\
&\quad - \sum_{\theta=1}^{t} \sum_{j=1}^{n} T_{ijp\theta} - \sum_{\theta=1}^{t} d_{ip\theta} - (\Gamma_{ipt} \times v_{ipt}) - \sum_{\theta=1}^{t} g_{ip\theta t} \Bigg) \\
&\quad \times (1-\alpha) \times -\pi'_p
\end{aligned}
$$

$$\times\left(Inv0+\sum_{\theta=1}^{t}\sum_{k=1}^{K}q_{ipk\theta}+\sum_{\theta=1}^{t}\sum_{j=1}^{n}T_{jip\theta}-\sum_{\theta=1}^{t}\sum_{j=1}^{n}T_{ijp\theta}\right.$$
$$\left.-\sum_{\theta=1}^{t}d_{ip\theta}-(\Gamma_{ipt}\times v_{ipt})-\sum_{\theta=1}^{t}g_{ip\theta t}\right);\qquad \forall i\in I,p\in P,t\in T;t\neq T \tag{3.30}$$

$$R_{ipt}\geq-\pi_p'\times\left(Inv0+\sum_{\theta=1}^{t}\sum_{k=1}^{K}q_{ipk\theta}+\sum_{\theta=1}^{t}\sum_{j=1}^{n}T_{jip\theta}\right.$$
$$\left.-\sum_{\theta=1}^{t}\sum_{j=1}^{n}T_{ijp\theta}-\sum_{\theta=1}^{t}d_{ip\theta}-(\Gamma_{ipt}\times v_{ipt})-\sum_{\theta=1}^{t}g_{ip\theta t}\right);\qquad \forall i\in I,p\in P,t\in T;t=T \tag{3.31}$$

$$I_{0pt}\leq u_{0p};\qquad \forall p\in P,t\in T \tag{3.32}$$

$$\left(Inv0+\sum_{\theta=1}^{t}\sum_{k=1}^{K}q_{ipk\theta}+\sum_{\theta=1}^{t}\sum_{j=1}^{n}T_{jip\theta}-\sum_{\theta=1}^{t}\sum_{j=1}^{n}T_{ijp\theta}\right.$$
$$\left.-\sum_{\theta=1}^{t}d_{ip\theta}+(\Gamma_{ipt}\times v_{ipt})+\sum_{\theta=1}^{t}g_{ip\theta t}\right)\leq u_{ip};\qquad \forall i\in I,p\in P,t\in T \tag{3.33}$$

$$\sum_{j=1}^{n}T_{ijpt}\leq\max\left(\left(Inv0+\sum_{\theta=1}^{t}\sum_{k=1}^{K}q_{ipk\theta}+\sum_{\theta=1}^{t}\sum_{j=1}^{n}T_{jip\theta}\right.\right.$$
$$-\sum_{\theta=1}^{t}\sum_{j=1}^{n}T_{ijp\theta}-\sum_{\theta=1}^{t}d_{ip\theta}+(\Gamma_{ipt}\times v_{ipt})$$
$$\left.\left.+\sum_{\theta=1}^{t}g_{ip\theta t}\right),0\right)+\sum_{k=1}^{K}q_{ipkt};\qquad \forall i\in I,p\in P,t\in T;i\neq j \tag{3.34}$$

$$v_{ipt}+g_{ip\theta t}\geq\hat{d}_{ipt};\qquad \forall i\in I,p\in P,t\in T,\theta\leq t \tag{3.35}$$

(3.2), (3.5)−(3.8), (3.10)−(3.17)

$$q_{ipkt},P_{pt},I_{ipt},T_{ijpt},\mathrm{LB}_{ipt},\mathrm{OP}_{pt},v_{ipt},g_{ip\theta t},w_{ikt},$$
$$R_{ipt},v_{ipt},g_{ip\theta t}\geq 0;\qquad \forall\{i,j\}\in I,p\in P,t\in T,k\in K,\theta\leq t \tag{3.36}$$

$$x_{ijkt},z_{ikt},y_{pt}\in\{0.1\};\qquad \forall i\in I_0,p\in P,t\in T,k\in K. \tag{3.37}$$

The inventory and shortage cost for each retailer will be calculated through the constraints (3.29)–(3.31). Then, the inventory holding capacity of each retailer is respected by constraints (3.33) using the new inventory calculation method. Since in the new approach, we calculate the amount of inventory and shortage in one equation; it is necessary to use the maximization function in constraints (3.34) to allow periods with positive inventory doing transshipment.

$$\sum_{j=1}^{n}T_{ijpt}\leq lin_{ipt}+\sum_{k=1}^{K}q_{ipkt};\qquad \forall i\in I,p\in P,t\in T;i\neq j \tag{3.38}$$

$$lin_{ipt}\geq\left(Inv0+\sum_{\theta=1}^{t}\sum_{k=1}^{K}q_{ipk\theta}+\sum_{\theta=1}^{t-1}\sum_{j=1}^{n}T_{jip\theta}-\sum_{\theta=1}^{t-1}\sum_{j=1}^{n}T_{ijp\theta}\right.$$

$$-\sum_{\theta=1}^{t-1} d_{ip\theta} + (\Gamma_{ipt} \times v_{ipt}) + \sum_{\theta=1}^{t-1} g_{ip\theta t}\Bigg); \qquad \forall i \in I, p \in P, t \in T; i \neq j \tag{3.39}$$

$$lin_{ipt} \geq 0; \qquad \forall i \in I, p \in P, t \in T \tag{3.40}$$

$$lin_{ipt} \geq M \times linY_{ipt}; \qquad \forall i \in I, p \in P, t \in T \tag{3.41}$$

$$\begin{aligned} lin_{ipt} \leq \Bigg(& Inv0 + \sum_{\theta=1}^{t}\sum_{k=1}^{K} q_{ipk\theta} + \sum_{\theta=1}^{t-1}\sum_{j=1}^{n} T_{jip\theta} - \sum_{\theta=1}^{t-1}\sum_{j=1}^{n} T_{ijp\theta} \\ & -\sum_{\theta=1}^{t-1} d_{ip\theta} + (\Gamma_{ipt} \times v_{ipt}) + \sum_{\theta=1}^{t-1} g_{ip\theta t}\Bigg) + M \times (1 - linY_{ipt}); \qquad \forall i \in I, p \in P, t \in T; i \neq j \end{aligned} \tag{3.42}$$

$$linY_{ipt} \in \{0, 1\}; \qquad \forall i \in I, p \in P, t \in T. \tag{3.43}$$

We linearized the maximization function in constraints (3.34) by defining two auxiliary variables, *i.e.*, lin_{ipt} and $linY_{ipt}$.

4. Resolution of the problem

4.1. Computational complexity

It has been proven that the LSP with production capacity and the VRP are NP-Hard problems [3, 38]. Therefore, combining these two problems with lateral transshipment, time windows, production setup time, and outsourcing will also be NP-Hard. Solving these NP-Hard problems is time-consuming, so it is recommended to use heuristic or matheuristic methods to find near-optimal solutions.

4.2. Proposed algorithms

Two well-known metaheuristic algorithms, a GA and a SA algorithm, have been developed to solve this problem, the former known as population-based and the latter as individual-based. The reason for using different algorithms is to provide a useful comparison basis for a large-size set of problems the CPLEX solver usually fails to solve. Moreover, parameter tuning for all metaheuristic algorithms presented in this article is done by applying the Taguchi method [39] (see Part A of the online supplement).

4.2.1. Genetic Algorithm (GA)

The GA is used as the first optimization algorithm in this article. The GA was first introduced by Fraser [40]. This algorithm is inspired by natural evolution to model the search method. Notably, one of the most essential and significant features of the GA is the parallel search for problem solution space. The basic steps and elements in the GA are the initial population, crossover and mutation operators, next population selection strategy, and stopping criteria. In this section, all these steps will be comprehensively described.

Chromosome representation. To represent a chromosome in the proposed GA, we use four-dimensional and two two-dimensional matrices that indicate the possibility of transshipment between two retailers, production in a specific period, and vehicle routes in that period. A sample chromosome is illustrated in Figure 1.

Initial population. Given that the GA is not highly sensitive to the initial population, we generated the initial population randomly while ensuring that the generated solutions meet certain conditions that prevent infeasible solutions. It is important to note that we took into account the time windows during the generation of the initial population (as shown in Algorithm 1), which means that the related constraints will be automatically incorporated into the mathematical model. We effectively remove their associated constraints from the mathematical model by considering the time windows simultaneously with generating the initial population. This approach simplifies the problem and reduces the algorithm's complexity, making it more efficient and effective in finding feasible solutions.

	t_0	t_1	t_2
p_1	0	1	0
p_2	0	0	1
p_3	0	0	1

(A)

t	Routs				
t_0	-	-	-	-	-
t_1	0	1	0	2	3
t_2	0	2	0	1	-

(B)

$t = 1$, $p = 2$	i_0	i_1	i_2	i_3
i_0	0	0	0	0
i_1	0	0	0	0
i_2	0	0	0	1
i_3	0	0	0	0

(C)

FIGURE 1. Chromosomes representation. (a) Possibility of production. (b) Vehicle routs. (c) Possibility of transshipment.

Algorithm 1. Considering time windows for initial population.

1: V_k is defined as a set of retailers which are assigned to vehicle k and let $k \leftarrow 0$ and Penalty Factor $\leftarrow 1$.
2: **if** $k = K$ **then** the algorithm will be terminated
3: **else** $k \leftarrow k + 1$ go to step 4.
4: Let $i \leftarrow 0$
5: **if** $\mathrm{V}_k(i) = |\mathrm{V}_k|$ **then** go to step 2
6: **else** Set $i \leftarrow i + 1$.
7: **if** $i = 1$ **then** Compute $w_{itk} =$ Service start time at the factory $+ \tau_{0ik}$ and go to step 9
8: **if** $i \neq 1$ **then** Compute $w_{itk} = w_{(i-1)tk} + \tau_{(i-1)ik}$ and go to step 9
9: **if** $(e_{it} > z_{ikt} \,||\, l_{it} < z_{ikt} \times w_{ikt})$ **then** let Penalty Factor $\leftarrow 10^6$; Stop
10: **else** Go to Step 5

Parent selection. For both crossover and mutation operators, we consider three types of parent selection methods in the model: random selection, tournament selection, and roulette wheel selection. Based on the results of GA implementation in several identical examples, roulette wheel selection performs better than the other two methods. Algorithm 2 shows the pseudo-code of the parent selection procedure, where $nPop$ represents the population size, $Costs$ and $WorstCost$ are matrices for the total cost of population and worst-case population cost, respectively, p_i represents parent i, and the ith member of the population is represented by $pop(i)$.

Algorithm 2. Parent selection procedure.

1: There are 3 options: "Random selection", "Roulette wheel selection", and "Tournament selection".
2: Set $SelectionPressure \leftarrow Beta$
3: **if** (the first option is chosen) **then** go to step 6.
4: **if** ("Roulette wheel selection" is selected) **then** go to step 8.
5: **if** (the last one is taken) **then** go to step 11.
6: Generate two random numbers between $[1, nPop]$ and then assign them to i_1 and i_2, respectively.
7: Let $p_1 = pop(\mathrm{i}_1)$ and $p_2 = pop(i_2)$; Stop
8: For $(i = 1, nPop, i++)$: Compute: $P(i) = e^{\frac{-SelectionPressure \times Costs(i)}{WorstCost}}$ and then calculate $P(i) = \frac{p(i)}{\sum_i P(i)}$.
9: Generate two random numbers r_1 and r_2 between 0 and 1.
10: For $(i = 1, nPop, i++ \,\&\&\, j = 1, j \leq 2, j++)$: **if** $r_j \leq \sum_{i \in nPop} P(i)$ **then** $p_j = pop(i)$; Stop.
11: For $(j = 1, j \leq 2, j++)$: Select m member of the population and then assign the best one to the p_j; Stop.

Crossover operator. We present single-point crossover [41], double-point crossover [42], and uniform crossover [43] to combine genetic information of two selected parents and generate new offspring.

Mutation operator. Trapping in local optimum is one of the challenging issues in GAs, which is prevented by applying mutation operators.

Next population selection strategy. We use equation (4.1) to determine the number of members selected from the best chromosomes, which increases as the algorithm progresses to choose the next generation members (Algorithm 3).

$$p_{\text{elitism}} = ip_{\text{elitism}} + (1 - ip_{\text{elitism}}) \times e^{\left(1 - \frac{MaxIteration}{Iteration}\right)}. \tag{4.1}$$

Algorithm 3. New generation selection procedure.

1: Sort current generation and new chromosomes according to their objective function value.
2: For $(i = 1, i \leq (p_{\text{elitism}} \times popsize), i++)$: Transfer chromosome i in sorted list to new generation .
3: Remove all conducted chromosome from sorted list
4: For $(i = \lceil p_{\text{elitism}} \rceil, i \leq popsize, i++)$
5: {
6: Generate a random number between 1 and number of chromosomes in sorted list
7: Select item in position which is specified in the previous step and move it to the new generation
8: Remove selected item from sorted list (Current Generation)
9: }

Restarting procedure. If no improvement in the objective function's value is seen after $NotImprove$ iteration, the algorithm will again generate initial feasible solutions. It will eventually maintain a particular percentage of the best available solutions and replace others with the recently generated population (Algorithm 4).

Algorithm 4. Restarting procedure.

1: **if** (GA's fitness function does not have any improves for a predefined number of iterations) **then** go to step 2.
2: Save α% of best chromosomes in new generation.
3: For $(i = \alpha \times Popsize, i \leq Popsize, i++)$
4: {
5: Generate Initial Solution for production setup, routing and transshipment
6: Add new chromosomes to the new generation
7: }

Termination criterion. We set a maximum number of iterations for the GA, referred to as $MaxIteration$, to terminate the algorithm.

4.2.2. Simulated Annealing algorithm (SA)

We applied the SA as the second metaheuristic algorithm in this paper. This algorithm was first proposed by Metropolis *et al.* [44]. The SA originated from the physical process of cooling solid objects. In other words, this concept establishes a link between thermodynamics and the search to find a globally optimal or near-optimal solution in different optimization problems. We explain the steps of this algorithm in more detail below.

Initial solution. In the SA, the initial solution is generated randomly, while in the modified SA, explained in Section 4.2.3, we create the initial solution based on a heuristic approach.

Initial and final temperatures. In the proposed SA algorithm, obtaining the worst and best possible solutions to determine the initial and final temperatures is necessary. To calculate the worst possible solution, it is assumed

(A) (B) (C)

FIGURE 2. Different neighborhood search procedure. (a) Swap. (b) Reversion. (c) Insertion.

that there is no production at the factory, and all products must be outsourced to prevent shortages. On the other hand, to obtain the best solution, we relax the binary constraints of the model and allow the model to choose between factory production, outsourcing, and shortages in each period by itself. This solution is the best but an infeasible solution of the model and is used to calculate the final temperature. Overall, obtaining the worst and best possible solutions helps determine the range of possible solutions and allows the SA algorithm to explore the solution space effectively.

Temperature reduction procedure. The geometric decrement scheme (Eq. (4.2)) is used to design the temperature reduction process. In this method, at each step of temperature reduction, the previous temperature is multiplied by the factor α, which is less than one, to calculate the new temperature.

$$\mathrm{Temp}_k = \alpha \times \mathrm{Temp}_{(k-1)}. \tag{4.2}$$

Neighborhood search. Three methods, including swap, reversion, and insertion, have been designed and applied for neighborhood search, as shown in Figure 2. In each step, one of them is selected randomly.

Accept/Decline function. We use the common function for minimization problems presented in equation (4.3).

$$p(\text{accept as a neighbor}) = \begin{cases} e^{-\frac{f(s')-f(s)}{t_k}}, & f(s')-f(s) > 0; \\ 1, & f(s')-f(s) \leq 0. \end{cases} \tag{4.3}$$

$f(s')$ is the objective function value for the new solution (s') in the neighborhood of s and (t_k) represents the current temperature.

Termination criterion. We terminate the SA after reaching the final temperature.

4.2.3. Modified SA (MSA)

The MSA differs from the SA in two ways:

(1) The Proposed procedure for generating the initial solution.
(2) Neighborhood Search.

Initial solution for production setup variable. First, we calculate the total demand for all types of products in each period. In case of demand uncertainty, we consider their average. Next, we determine the production capacity of the plant in each period. This can result in one of the following two scenarios:

(1) The aggregate demand is greater than or equal to the production capacity, which then we have to utilize the full capacity of the plant and, if necessary, consider outsourcing options.
(2) The aggregate demand is less than the production capacity, which in this case, all y_{pt} members will not take the value one. Still, according to the capacity of the factory in each period, only a few cells will get a value of one to meet all demands.

Initial solution for routing variables. First, we randomly assign retailers to each vehicle to generate the initial solution for the routing variables. Then, we determine the sequence of visiting retailers for each vehicle based on the shortest path. This algorithm is also known as a greedy algorithm, which usually does not achieve the optimal solution because it only takes one step at a time. However, it is suitable for generating an initial solution for the SA.

Neighborhood search. In the SA, we generate a solution and then use one of the three methods described in Section 4.2.2 for neighborhood searching. This process is repeated only once to produce the next solution. However, in the MSA, we generate $nMove$ neighbors for the current solution, calculate the target function for each, and choose the best neighbor among them. This differs from SA, where only one neighbor is generated, and the Accept/Decline function is called.

5. Computational results

This section discusses the importance of information transparency in a supply chain. Then, we evaluate the performance of the three algorithms developed in this study and analyze the impact of transshipment, a new concept in PRPs. Additionally, we consider the ratio of outsourcing cost to production cost and the level of decision-makers' conservatism. To enhance our numerical analysis, we include supplementary content (see attachment) that offers further insights into:

– The Value of Integrated Decisions in the Supply Chain (Part B).
– Parameters used for evaluating the algorithms (Part C).

5.1. The value of information transparency

Based on an article by Gardner *et al.* [45], transparency in the supply chain means that all relevant information is readily available and easily understood. Transparency provides many benefits to the supply chain, such as enabling companies to ensure that customer and supplier demands are genuine. This allows companies to manufacture and supply their products or services more confidently. In this paper, when we refer to information transparency, we mean having reliable information about customer demands. When transparency is lacking, demands cannot be determined, and decisions must be made based on their means and variations. We take different amounts of demand variability into account for five various examples to see the effects of this phenomenon. Then we calculate the supply chain cost and present them in Figure 3. The figure shows that as information transparency decreases, demand variability increases, and consequently, supply chain costs increase significantly. The trend continues to grow with volatility. Therefore, managers should share information throughout the supply chain, make more precise forecasts, sign more appropriate contracts, etc., to increase information transparency and achieve cost savings in the long term.

5.2. Effectiveness of proposed solution approaches

We used fifteen numerical examples based on Farghadani-Chaharsooghi *et al.* [17] to investigate the effectiveness of the proposed solution approaches. We tested both the GA and the MSA in the DM, as well as all three solution approaches (GA, SA, and MSA) in the SM. We solved these examples (where possible) exactly using the Cplex solver to provide a solid basis for comparing the proposed algorithms. All numerical examples were executed on a personal computer with the following specifications: Intel Core i7 2.20 GHz and 6.00 GB RAM with Windows 8.1.

5.2.1. Numerical results of the GA and the MSA for the DM

The number of periods has been selected in the range of 3–12, and the total number of retailers along with the production factory is also taken in the range of 6–50 numbers to obtain different dimensions of the problem. Part C of the online supplement presents the parameters used to analyze the proposed solution approaches' efficiency. The results of this implementation are presented in Tables A.1 and A.2. The columns for GA-A-GAP, GA-B-GAP, MSA-A-GAP, and MSA-B-GAP in these tables are calculated by equations (5.1)–(5.4), respectively.

$$\text{GA-A-GAP} = \frac{\text{Average Objective Function}_{\text{GA}} - \text{Objective Function}_{\text{Cplex}}}{\text{Objective Function}_{\text{Cplex}}} \times 100 \tag{5.1}$$

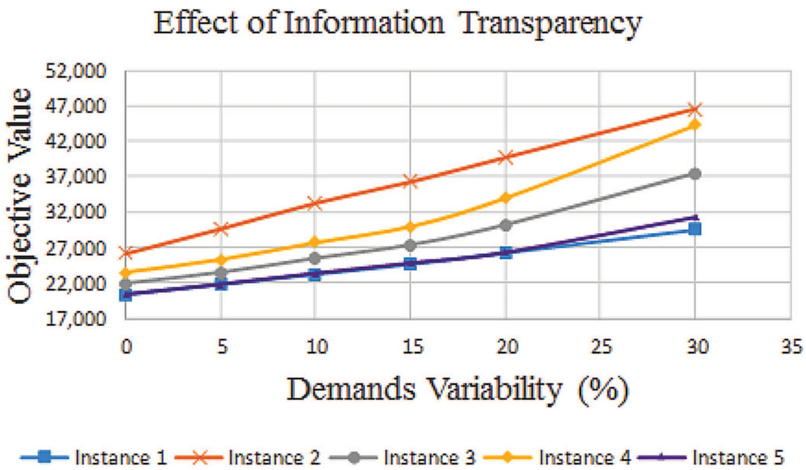

FIGURE 3. Impacts of information transparency on supply chain total cost.

$$\text{GA-B-GAP} = \frac{\text{Best Objective Function}_{\text{GA}} - \text{Objective Function}_{\text{Cplex}}}{\text{Objective Function}_{\text{Cplex}}} \times 100 \tag{5.2}$$

$$\text{MSA-A-GAP} = \frac{\text{Average Objective Function}_{\text{MSA}} - \text{Objective Function}_{\text{Cplex}}}{\text{Objective Function}_{\text{Cplex}}} \times 100 \tag{5.3}$$

$$\text{MSA-B-GAP} = \frac{\text{Best Objective Function}_{\text{MSA}} - \text{Objective Function}_{\text{Cplex}}}{\text{Objective Function}_{\text{Cplex}}} \times 100. \tag{5.4}$$

For large-scale problems where the Cplex Solver failed to achieve an optimal solution, we used the lower bounds obtained by the software for each problem for comparison.

5.2.2. Comparison and quality analysis of two proposed algorithms for the DM

Overall, the numerical results of implementing both metaheuristics on the DM show that both algorithms provide reasonable solutions within a reasonable amount of time. However, on average, the GA solutions have a smaller gap compared to the MSA. This could be due to the fact that the GA is a population-based algorithm that compares and combines several solutions in each iteration, while the SA does not have this ability.

5.2.3. Numerical results and quality analysis of the GA, the SA, and the MSA for the SM

In this part, the same process described in the second part is used, and the results are presented in Tables A.3–A.5. It is worth noting that when comparing the quality of solutions, the GA algorithm outperformed the other two proposed algorithms once again. However, it is also clear that the modifications made to the SA algorithm to obtain the MSA algorithm were effective, as they reduced the average gap by approximately 2%.

5.3. Effects of considering lateral transshipment option in PRPs

We conducted two sets of experiments, one with lateral transshipment allowed between retailers and the other without it, to examine its effects on the supply chain. Table 2 presents the results of solving several examples in both scenarios. When lateral transshipment is not allowed, the total cost incurred by the supply chain is always higher than when retailers can obtain items from other retailers within the same echelon. Therefore, considering the possibility of lateral transshipment can result in significant cost savings. Another interesting observation is that the supply chain becomes less willing to use trucks to transport goods from the factory to retailers as the fixed transportation cost increases. Instead, the model attempts to use the fewest number of trucks possible and maximize their capacity utilization. Therefore, it sends items from the factory to the most

TABLE 2. The effect of fixed cost of transportation on the lateral transshipment.

# Instances	Fixed transportation cost	Objective values		Transshipment cost	Effectiveness of transshipment (%)
		Without transshipment	With transshipment		
1	30	140 283.67	137 982.67	3220.00	1.64
2	40	141 561.67	139 155.67	3730.00	1.70
3	50	142 189.67	139 732.67	4150.00	1.73
4	60	144 415.67	141 812.67	4260.00	1.80
5	70	144 790.00	141 822.67	4335.00	2.05

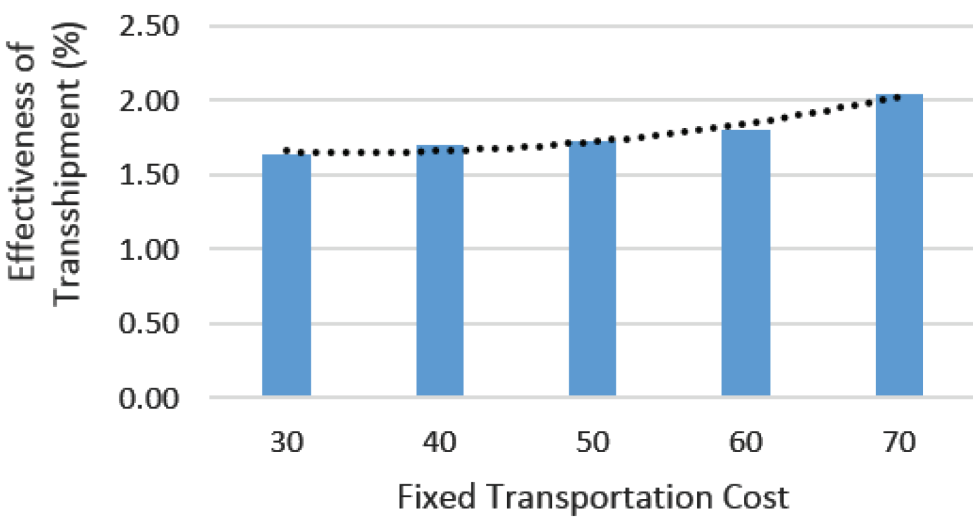

FIGURE 4. The effectiveness of lateral transshipment based on fixed transportation cost.

suitable retailer regarding location and maintenance capacity, allowing for lateral transshipment in subsequent periods. This observation is supported by Figure 4 and Table 2.

5.4. Outsourcing *vs.* production

As this study considers the outsourcing of products in the PRPs for the first time, it is important to investigate the effects of unit outsourcing cost on the factory's production rate and the number of outsourced goods in each period. For this purpose, we consider a problem with five periods, five nodes, two products, and two vehicles and solve it for different ratios of the unit outsourcing cost to the unit production cost. The results are presented in Figure 5. According to Figure 5, if the unit outsourcing cost is equal to the variable production cost (Ratio one), outsourcing all products in all periods is recommended since production in the factory imposes an undesirable fixed cost on the manufacturer. This trend will continue for smaller ratios than one. However, outsourcing will reduce when the unit cost of outsourcing exceeds the unit production cost in the manufacturing plant. There will be a dramatic decrease in outsourcing for ratios greater than 5. For ratios greater than 7, the outsourcing rate will be zero. Hence factory production in each period can have a certain amount due to limited resources. With the reduction of outsourcing, the shortage will increase, and in turn, the maintenance and transportation costs will reduce.

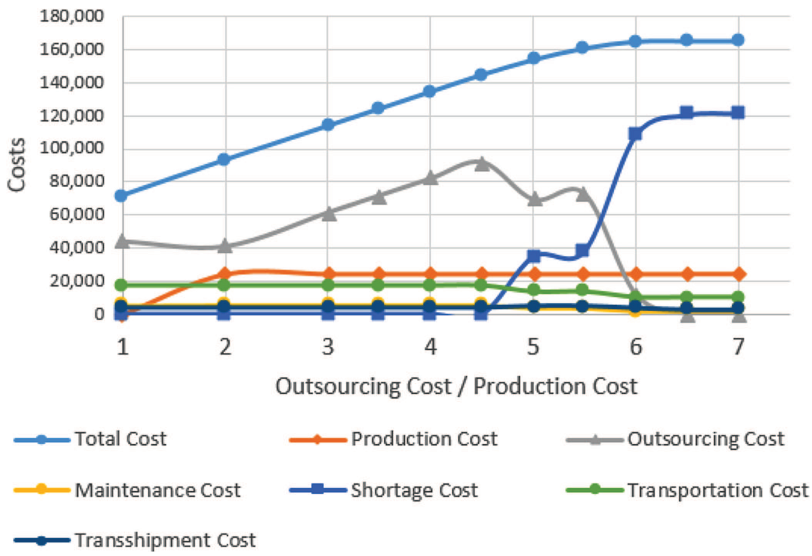

FIGURE 5. The effect of unit outsourcing cost to unit production cost ratio.

5.5. Impacts of decision-makers' conservatism

We conducted sensitivity analysis experiments by varying the degree of conservatism used to evaluate the sensitivity of the objective function to demand fluctuations, as obtained from equation (5.5).

$$\hat{d}_{ipt} = \gamma \times d_{ipt} \tag{5.5}$$

where d_{ipt} indicates the nominal values of the demand, and γ represents the percentage change in the demand parameter. It is vital to define the degree of conservatism as a linear function based on time [46]. In equation (5.6), m and k take a positive value and a value in the range of $[0, 1]$ respectively to avoid very conservative choices.

$$\Gamma_{ipt} = kt + m. \tag{5.6}$$

In previous studies on robust optimization, five relationships have been commonly considered for the degree of conservatism, which are presented in equations (5.7)–(5.11):

$$\text{Scenario 1}: \quad \Gamma_{ipt} = 0.05t + 0.1 \tag{5.7}$$

$$\text{Scenario 2}: \quad \Gamma_{ipt} = 0.10t + 0.5 \tag{5.8}$$

$$\text{Scenario 3}: \quad \Gamma_{ipt} = 1 \tag{5.9}$$

$$\text{Scenario 4}: \quad \Gamma_{ipt} = 0.50t + 0.5 \tag{5.10}$$

$$\text{Scenario 5}: \quad \Gamma_{ipt} = t. \tag{5.11}$$

Figure 6 shows how the cumulative protection of uncertainty budgets works during the planning horizon. For instance, when there is a 12-period planning, and the fourth scenario sets the uncertainty budget, the minimum protection is 1, and the maximum is 6.5. This implies that the model will be entirely protected in the first period, whereas in the last period, it is only protected against fluctuations of 54% of the coefficients. The worst-case scenario or Soyster's Approach [47] is the last scenario, where all periods will always be protected against all fluctuations.

Figure 7 demonstrates the probability of violating the constraints based on inequality (50) in Jabbarzadeh *et al.* [37] for different degrees of conservatism. As expected, increasing the degree of conservatism against

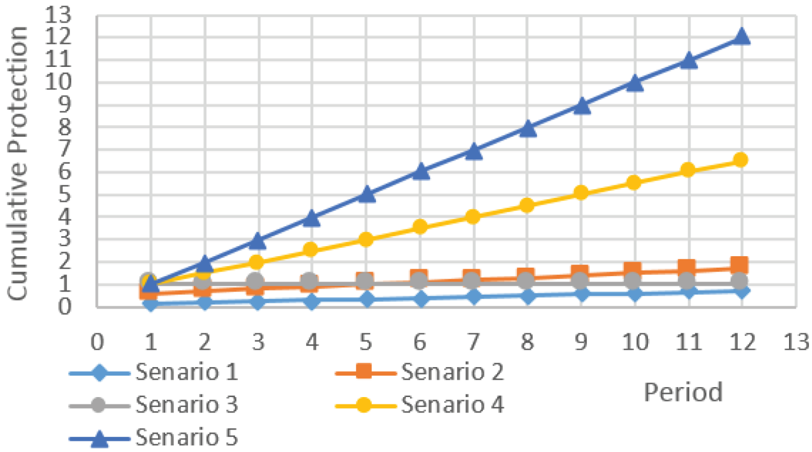

FIGURE 6. The cumulative protection of the budgets of uncertainty.

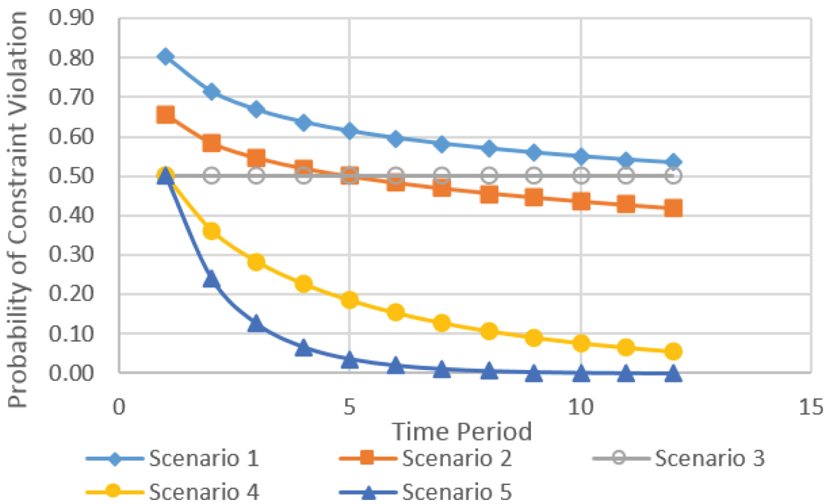

FIGURE 7. Possibility of constraints' violation for five different degrees of conservatism.

demand uncertainty reduces the probability of constraint violation. Additionally, it is observed that the probability of violating the limitations in the fifth scenario decreases more quickly and tends to reach zero after a large number of periods. In other words, the last scenario is the most conservative, as after nearly eight periods, all its constraints will always be feasible. In contrast, this value is more than 12 periods for the fourth scenario. Scenarios 1, 2, and 3 also have poor performance, even after a long period. To achieve a violation probability of less than 0.1 in scenarios 1 and 2, roughly 693 and 175 periods must be passed, respectively.

Next, in Figure 8, we analyze the impact of demand fluctuations and the degree of conservatism on the objective function values. We first calculate the objective function values based on average demands when decision-makers have complete information about demands. Then, we compare it to the supply chain cost when there is no transparency in information. Figure 8 illustrates that the greater the demand fluctuations for each scenario, the higher the percentage difference between the deterministic and robust objective values. Additionally, it shows that the more conservative the scenario was chosen to address demand uncertainty, the

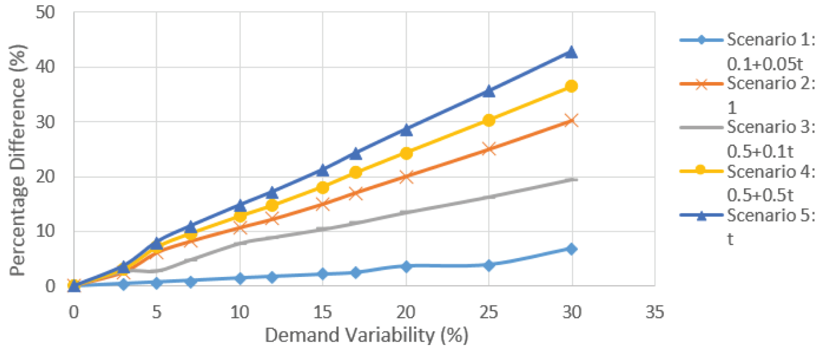

Figure 8. Difference between deterministic and robust objective values in percentage.

greater the increase in cost. This increase in cost in more conservative scenarios is commonly referred to in the literature as the Price of Robustness.

To further evaluate the performance of the robust approach proposed in this study, we generated 100 random examples [36, 37, 47]. Demands were uniformly considered in the range of $[d_{ipt}, d_{ipt} + \bar{d}_{ipt}]$. To compare the results, we first solved the deterministic model by considering demands as much as possible $(d_{ipt} + \bar{d}_{ipt})$ and calculated the supply chain cost. Then, we executed the robust model with different degrees of conservatism and considered the demand in the ranges d_{ipt} to $(d_{ipt} + \bar{d}_{ipt})$. Table B.1 shows the percentage difference of the average and standard deviation of supply chain costs in two deterministic and robust approaches for various degrees of conservatism and demand fluctuation rates. It also compares the best and worst solutions in the two approaches. For instance, according to Table B.1, when there is a 15% demand fluctuation and a robust model with $\Gamma_{ipt} = 0.1t + 0.5$ is developed and used, an average improvement of 15.62% in supply chain costs will be achieved. Additionally, the standard deviation of the robust model's solutions is 19.83% less than the DM. Furthermore, in the proposed case, the best and worst solutions identified in the deterministic model impose approximately 12.18% and 13.68% more costs on the supply chain than the robust approach, respectively. In general, according to Table B.1, the robust approach presented in this study outperforms the deterministic model (worst-case scenario) in all degrees of freedom and demand fluctuations.

6. Conclusions

In this study, we investigated the impact of incorporating lateral transshipment into PRP (Plant Location Problem) models while also considering outsourcing, hard time windows, and demand uncertainty. To achieve this, we developed a mixed-integer linear program that includes all the proposed features and further created a robust mathematical model to address demand uncertainty through five different scenarios. Since this problem is known to be NP-hard, we had to find suitable solution approaches to solve it within a reasonable time. To accomplish this, we employed three metaheuristic algorithms: GA, SA, and MSA. The computational results showed that all three algorithms were effective and efficient. However, GA performed best due to its population-based nature, unlike the other two individual-based algorithms.

In the sensitivity analysis section, we first showed that incorporating lateral transshipment into PRPs yields profits. Next, we conducted experiments on instances with different levels of demand uncertainty and solved them using various scenarios to analyze their impacts. Ultimately, we provide supply chain managers with the flexibility to choose a scenario based on their personality traits and level of conservatism. This empowers them to make informed decisions that align with their values and preferences. From a research perspective, one can consider uncertain vehicle travel times based on road traffic during different day hours. Besides, studying the possibility of various disruptions, such as accidents, is reasonable. Furthermore, developing a model that considers perishable products, GHG emissions, and workforce balancing with simultaneous pickup and delivery is recommended for future works.

Appendix A. Numerical results

Table A.1. Numerical results of the implementation of the GA on the DM.

General information for each instance					GA			Cplex		GA-A -GAP (%)	GA-B -GAP (%)
# Problem	# Period	# Nodes	# Products	# Vehicles	Average objective function	Best objective function	Execution time (h:mm:ss)	Objective function	Cplex time (h:mm:ss)		
1	3	6	2	2	70 689	70 486	0:01:57	70,486	0:02:14	0.29	0.00
2	3	6	3	3	110 504	109 594	0:02:04	109,594	0:12:11	0.83	0.00
3	3	7	2	3	96 489	96 354	0:02:20	96,354	1:59:59	0.14	0.00
4	4	4	2	2	57 958	57 435	0:02:19	57,435	0:03:21	0.91	0.00
5	4	5	2	2	84 698	84 445	0:02:07	84,445	0:03:53	0.30	0.00
6	4	6	2	2	120 097	119 495	0:01:58	119,245	0:40:35	0.71	0.21
7	5	5	2	2	133 713	132 385	0:00:55	131,350	0:18:43	1.80	0.79
8	5	6	3	3	276 692	275 795	0:05:59	272,832	0:23:24	1.42	1.09
9	5	7	3	3	395 236	381 026	0:07:02	379,818	1:59:59	4.06	0.32
10	6	10	3	4	1 052 564	1 028 276	0:07:34	1,017,379	1:59:59	3.46	1.07
11	7	15	4	5	4 205 562	4 189 511	0:10:23	4,171,576	1:59:59	0.81	0.43
12	8	18	5	6	9 309 058	9 276 137	0:15:29	9,257,875	1:59:59	0.55	0.20
13	9	20	6	8	14 120 572	14 030 969	0:22:44	N/A*	1:59:59	–	–
14	11	40	5	8	15 947 361	15 872 870	0:48:18	N/A*	1:59:59	–	–
15	12	50	5	10	49 655 608	49 486 440	1:22:27	N/A*	1:59:59	–	–

Table A.2. Numerical results of the implementation of the MSA on the DM.

General information for each instance					MSA			Cplex		MSA-A -GAP (%)	MSA-B -GAP (%)
# Problem	# Period	# Nodes	# Products	# Vehicles	Average objective function	Best objective function	Execution time (h:mm:ss)	Objective function	Cplex time (h:mm:ss)		
1	3	6	2	2	70 708	70 597	0:03:24	70 487	0:02:14	0.31	0.16
2	3	6	3	3	110 513	109 870	0:04:58	109 595	0:12:11	1.83	0.49
3	3	7	2	3	98 055	97 951	0:05:33	96 354	1:59:59	1.14	0.83
4	4	4	2	2	58 654	58 160	0:02:58	57 435	0:03:21	0.91	0.77
5	4	5	2	2	85 188	84 825	0:02:44	84 445	0:03:53	0.88	0.45
6	4	6	2	2	120 767	119 995	0:02:35	119 245	0:40:35	1.28	0.63
7	5	5	2	2	133 107	132 550	0:02:38	131 350	0:18:43	1.34	0.91
8	5	6	3	3	283 571	277 647	0:05:25	272 832	0:23:24	1.42	1.09
9	5	7	3	3	407 915	384 459	0:05:35	379 818	1:59:59	7.40	1.22
10	6	10	3	4	1 099 343	1 059 850	0:05:57	1 017 379	1:59:59	8.06	4.17
11	7	15	4	5	4 245 211	4 202 431	0:08:09	4 171 576	1:59:59	1.77	0.74
12	8	18	5	6	9 442 117	9 384 401	0:12:04	9 257 876	1:59:59	0.55	0.20
13	9	20	6	8	14 319 984	14 258 654	0:19:40	N/A	1:59:59	–	–
14	11	40	5	8	16 182 704	16 156 731	0:55:56	N/A	1:59:59	–	–
15	12	50	5	10	49 861 620	49 725 836	1:16:19	N/A	1:59:59	–	–

TABLE A.3. Numerical results of the implementation of the GA on the SM.

General information for each instance					GA			Cplex		GA-A -GAP (%)	GA-B -GAP (%)
# Problem	# Period	# Nodes	# Products	# Vehicles	Average objective function	Best objective function	Execution time (h:mm:ss)	Objective function	Cplex time (h:mm:ss)		
1	3	5	1	2	13 322	13 160	0:02:17	13 160	0:00:15	1.23	0.00
2	3	6	1	2	21 999	21 613	0:02:12	21 578	0:21:11	1.95	0.16
3	3	6	2	2	43 723	43 339	0:02:17	43 102	1:59:59	1.44	0.55
4	4	4	2	2	31 915	31 480	0:02:24	31 277	0:02:01	2.04	0.65
5	4	5	2	2	42 014	40 885	0:02:21	40 458	1:59:59	3.84	1.05
6	4	6	2	2	100 491	98 544	0:02:23	96 432	1:59:59	4.21	2.19
7	5	4	2	2	49 650	48 945	0:02:51	48 554	0:20:21	2.26	0.81
8	5	5	2	2	107 632	105 670	0:02:41	103 726	1:59:59	3.77	1.87
9	6	10	3	4	1 189 732	1 176 385	0:04:24	1 163 470	1:59:59	2.26	1.11
10	6	12	3	5	1 618 173	1 580 271	0:04:36	1 544 196	1:59:59	4.79	2.34
11	7	15	4	5	5 174 060	5 140 983	0:08:33	5 107 511	1:59:59	1.30	0.66
12	8	18	4	6	7 905 298	7 825 362	0:15:19	7 724 896	1:59:59	2.34	1.30
13	9	20	6	8	19 065 492	18 824 819	0:26:14	N/A	1:59:59	–	–
14	11	40	5	8	57 134 554	56 990 278	0:57:59	N/A	1:59:59	–	–
15	12	50	5	10	85 238 881	84 931 868	1:26:34	N/A	1:59:59	–	–

TABLE A.4. Numerical results of the implementation of the SA on the SM.

General information for each instance					SA			Cplex		SA-A -GAP (%)	SA-B -GAP (%)
# Problem	# Period	# Nodes	# Products	# Vehicles	Average objective function	Best objective function	Execution time (h:mm:ss)	Objective function	Cplex time (h:mm:ss)		
1	3	5	1	2	13 704	13 582	0:03:29	13 160	0:00:15	4.13	3.20
2	3	6	1	2	22 890	22 743	0:03:11	21 578	0:21:11	6.08	5.40
3	3	6	2	2	45 442	44 049	0:05:15	43 102	1:59:59	5.43	2.20
4	4	4	2	2	32 052	31 803	0:06:33	31 277	0:02:01	2.48	1.68
5	4	5	2	2	43 217	41 051	0:05:35	40 458	1:59:59	6.82	1.46
6	4	6	2	2	101 740	98 839	0:03:17	96 432	1:59:59	5.50	2.50
7	5	4	2	2	52 182	49 157	0:02:19	48 554	0:20:21	7.47	1.24
8	5	5	2	2	118 424	109 135	0:06:20	103 726	1:59:59	14.17	5.21
9	6	10	3	4	1 224 512	1 192 448	0:09:26	1 163 470	1:59:59	5.25	2.49
10	6	12	3	5	1 676 314	1 595 947	0:06:45	1 544 196	1:59:59	8.56	3.35
11	7	15	4	5	5 214 263	5 158 277	0:13:45	5 107 511	1:59:59	2.09	0.99
12	8	18	4	6	8 006 643	7 981 432	0:27:11	7 724 896	1:59:59	3.65	3.32
13	9	20	6	8	19 704 761	19 417 460	0:49:23	N/A	1:59:59	–	–
14	11	40	5	8	57 671 808	57 667 156	1:16:49	N/A	1:59:59	–	–
15	12	50	5	10	85 654 503	85 488 542	1:46:24	N/A	1:59:59	–	–

TABLE A.5. Numerical results of the implementation of the MSA on the SM.

General information for each instance					MSA			Cplex		MSA-A -GAP (%)	MSA-B -GAP (%)
# Problem	# Period	# Nodes	# Products	# Vehicles	Average objective function	Best objective function	Execution time (h:mm:ss)	Objective function	Cplex time (h:mm:ss)		
1	3	5	1	2	13 626	13 552	0:07:54	13 160	0:00:15	3.54	2.98
2	3	6	1	2	22 727	22 593	0:10:20	21 578	0:21:11	5.32	4.70
3	3	6	2	2	45 245	44 279	0:06:21	43 102	1:59:59	4.97	2.73
4	4	4	2	2	31 916	31 823	0:06:43	31 277	0:02:01	2.04	1.75
5	4	5	2	2	42 886	40 863	0:05:50	40 458	1:59:59	6.00	1.00
6	4	6	2	2	99 807	98 614	0:11:48	96 432	1:59:59	3.50	2.26
7	5	4	2	2	51 326	49 357	0:03:18	48 554	0:20:21	5.71	1.65
8	5	5	2	2	111 791	106 299	0:06:30	103 726	1:59:59	7.77	2.48
9	6	10	3	4	1 187 712	1 181 516	0:12:47	1 163 470	1:59:59	2.08	1.55
10	6	12	3	5	1 611 208	1 595 896	0:15:34	1 544 196	1:59:59	4.34	3.35
11	7	15	4	5	5 166 404	5 153 063	0:28:06	5 107 511	1:59:59	1.15	0.89
12	8	18	4	6	7 959 093	7 885 025	0:35:27	7 724 896	1:59:59	3.03	2.07
13	9	20	6	8	19 367 079	19 273 610	1:10:53	N/A	1:59:59	–	–
14	11	40	5	8	57 147 808	57 071 771	1:38:05	N/A	1:59:59	–	–
15	12	50	5	10	85 233 044	85 147 145	2:30:41	N/A	1:59:59	–	–

Appendix B. Effects of applying robust approach on the DM

TABLE B.1. Improving the performance of the robust model compared to the DM for different values of demand fluctuations and degrees of conservatism.

Measure	Demand variability (%)	Conservatism degree ($\Gamma_i pt$)			
		$0.05t + 0.1$	$0.1t + 0.5$	1	$0.5t + 0.5$
Average (%) {Standard Deviation (%)}	5	7.56 {8.75}	5.15 {3.44}	3.36 {1.32}	1.59 {0.19}
	10	15.49 {18.68}	10.8 {12.44}	6.65 {3.90}	3.31 {1.75}
	15	22.78 {30.37}	15.62 {19.83}	10.43 {10.43}	5.93 {8.14}
	20	29.49 {42.08}	20.63 {30.47}	13.54 {17.82}	6.91 {10.81}
	30	43.16 {60.34}	32.5 {47.90}	9.65 {25.76}	10.8 {14.33}
{Best Case (%); Worst Case (%)}	5	{3.98; 10.70}	{6.80; 6.01}	{2.16; 1.67}	{1.13; 0.53}
	10	{12.97; 19.51}	{7.76; 10.68}	{4.39; 3.29}	{2.25; 1.15}
	15	{19.06; 25.71}	{12.18; 13.68}	{6.38; 4.47}	{3.23; 2.30}
	20	{23.61; 31.21}	{14.62; 17.18}	{7.92; 5.64}	{4.10; 2.74}
	30	{30.52; 42.82}	{18.04; 26.34}	{9.90; 12.10}	{5.00; 8.98}

References

[1] H.M. Wagner and T.M. Whitin, Dynamic version of the economic lot size model. *Manage. Sci.* **5** (1958) 89–96.

[2] G.B. Dantzig and J.H. Ramser, The truck dispatching problem. *Manage. Sci.* **6** (1959) 80–91.

[3] Y. Adulyasak, J.-F. Cordeau and R. Jans, The production routing problem: a review of formulations and solution algorithms. *Comput. Oper. Res.* **55** (2015) 141–152.

[4] Y. Adulyasak, J.-F. Cordeau and R. Jans, Optimization-based adaptive large neighborhood search for the production routing problem. *Transp. Sci.* **48** (2014) 20–45.

[5] P. Chandra, A dynamic distribution model with warehouse and customer replenishment requirements. *J. Oper. Res. Soc.* **44** (1993) 681–692.

[6] P. Chandra and M.L. Fisher, Coordination of production and distribution planning. *Eur. J. Oper. Res.* **72** (1994) 503–517.

[7] C. Paterson, G. Kiesmüller, R. Teunter and K. Glazebrook, Inventory models with lateral transshipments: a review. *Eur. J. Oper. Res.* **210** (2011) 125–136.

[8] Y.T. Herer, M. Tzur and E. Yücesan, Transshipments: an emerging inventory recourse to achieve supply chain leagility. *Int. J. Prod. Econ.* **80** (2002) 201–212.

[9] J.J. Coyle, C.J. Langley, R.A. Novack and B. Gibson, Supply Chain Management: A Logistics Perspective. Nelson Education (2016).

[10] O. Kaya, Outsourcing vs. in-house production: a comparison of supply chain contracts with effort dependent demand. *Omega* **39** (2011) 168–178.

[11] D. Bertsimas and A. Thiele, A robust optimization approach to inventory theory. *Oper. Res.* **54** (2006) 150–168.

[12] N. Brahimi and T. Aouam, Multi-item production routing problem with backordering: a MILP approach. *Int. J. Prod. Res.* **54** (2016) 1076–1093.

[13] Y. Qiu, J. Qiao and P.M. Pardalos, A branch-and-price algorithm for production routing problems with carbon cap-and-trade. *Omega* **68** (2017) 49–61.

[14] N. Absi, C. Archetti, S. Dauzère-Pérès, D. Feillet and M.G. Speranza, Comparing sequential and integrated approaches for the production routing problem. *Eur. J. Oper. Res.* **269** (2018) 633–646.

[15] M. Chitsaz, J.-F. Cordeau and R. Jans, A unified decomposition matheuristic for assembly, production, and inventory routing. *INFORMS J. Comput.* **31** (2019) 134–152.

[16] A. Majidi, P. Farghadani-Chaharsooghi and S.M.J. Mirzapour Al-e Hashem, Sustainable pricing-production-workforce-routing problem for perishable products by considering demand uncertainty; a case study from the dairy industry. *Transp. J.* **61** (2022) 60–102.

[17] P. Farghadani-Chaharsooghi, P. Kamranfar, M.S. Mirzapour Al-e Hashem and Y. Rekik, A joint production-workforce-delivery stochastic planning problem for perishable items. *Int. J. Prod. Res.* **60** (2022) 6148–6172.

[18] I. Brekkå, S. Randøy, K. Fagerholt, K. Thun and S.T. Vadseth, The fish feed production routing problem. *Comput. Oper. Res.* **144** (2022) 105806.

[19] S.T. Vadseth, H. Andersson, M. Stålhane and M. Chitsaz, A multi-start route improving matheuristic for the production routeing problem. *Int. J. Prod. Res.* (2022) 1–22.

[20] Y. Adulyasak, J.-F. Cordeau and R. Jans, Formulations and branch-and-cut algorithms for multivehicle production and inventory routing problems. *INFORMS J. Comput.* **26** (2014) 103–120.

[21] I. Dayarian and G. Desaulniers, A branch-price-and-cut algorithm for a production-routing problem with short-life-span products. *Transp. Sci.* **53** (2019) 829–849.

[22] Y. Qiu, L. Wang, X. Xu, X. Fang and P.M. Pardalos, Formulations and branch-and-cut algorithms for multi-product multi-vehicle production routing problems with startup cost. *Expert Syst. App.* **98** (2018) 1–10.

[23] M. Darvish, C. Archetti and L.C. Coelho, Trade-offs between environmental and economic performance in production and inventory-routing problems. *Int. J. Prod. Econ.* **217** (2019) 269–280.

[24] Y. Qiu, M. Ni, L. Wang, Q. Li, X. Fang and P.M. Pardalos, Production routing problems with reverse logistics and remanufacturing. *Transp. Res. Part E: Logistics Transp. Rev.* **111** (2018) 87–100.

[25] C.M. Schenekemberg, C.T. Scarpin, J.E. Pecora Jr., T.A. Guimarães and L.C. Coelho, The two-echelon production-routing problem. *Eur. J. Oper. Res.* **288** (2021) 436–449.

[26] E.G. Manousakis, C.D. Tarantilis and E.E. Zachariadis, The cyclic production routing problem. *Int. J. Prod. Res.* (2023) 1–20.

[27] D. Hrabec, L.M. Hvattum and A. Hoff, The value of integrated planning for production, inventory, and routing decisions: a systematic review and meta-analysis. *Int. J. Prod. Econ.* **248** (2022) 108468.

[28] G. Laporte, Fifty years of vehicle routing. *Transp. Sci.* **43** (2009) 408–416.

[29] A.L. Soyster, Technical note – convex programming with set-inclusive constraints and applications to inexact linear programming. *Oper. Res.* **21** (1973) 1154–1157.

[30] J.M. Mulvey, R.J. Vanderbei and S.A. Zenios, Robust optimization of large-scale systems. *Oper. Res.* **43** (1995) 264–281.

[31] A. Ben-Tal and A. Nemirovski, Robust solutions of uncertain linear programs. *Oper. Res. Lett.* **25** (1999) 1–13.

[32] L. El Ghaoui, F. Oustry and H. Lebret, Robust solutions to uncertain semidefinite programs. *SIAM J. Optim.* **9** (1998) 33–52.

[33] D. Bertsimas and M. Sim, Robust discrete optimization and network flows. *Math. Program.* **98** (2003) 49–71.

[34] D. Bertsimas and M. Sim, The price of robustness. *Oper. Res.* **52** (2004) 35–53.

[35] K. Tong, F. You and G. Rong, Robust design and operations of hydrocarbon biofuel supply chain integrating with existing petroleum refineries considering unit cost objective. *Comput. Chem. Eng.* **68** (2014) 128–139.

[36] D. José Alem and R. Morabito, Production planning in furniture settings via robust optimization. *Comput. Oper. Res.* **39** (2012) 139–150.

[37] A. Jabbarzadeh, M. Haughton and F. Pourmehdi, A robust optimization model for efficient and green supply chain planning with postponement strategy. *Int. J. Prod. Econ.* **214** (2019) 266–283.

[38] B. Karimi, S.M.T. Fatemi Ghomi and J.M. Wilson, The capacitated lot sizing problem: a review of models and algorithms. *Omega* **31** (2003) 365–378.

[39] G. Taguchi, S. Chowdhury and Y. Wu, Taguchi's Quality Engineering Handbook. Wiley (2005).

[40] A.S. Fraser, Simulation of genetic systems by automatic digital computers II. Effects of linkage on rates of advance under selection. *Aust. J. Biol. Sci.* **10** (1957) 492–500.

[41] A. Hiassat, A. Diabat and I. Rahwan, A genetic algorithm approach for location-inventory-routing problem with perishable products. *J. Manuf. Syst.* **42** (2017) 93–103.

[42] Y.C. Huang, Enhanced genetic algorithm-based fuzzy multi-objective approach to distribution network reconfiguration. *IEE Proc. Gener. Transm. Distrib.* **149** (2002) 615–620.

[43] C. Skinner and P. Riddle, Expected rates of building block discovery, retention and combination under 1-point and uniform crossover, in Parallel Problem Solving from Nature – PPSN VIII, edited by X. Yao, E.K. Burke, J.A. Lozano, J. Smith, J.J. Merelo-Guervós, J.A. Bullinaria, J.E. Rowe, P. Tiňo, A. Kabán and H.-P. Schwefel. Springer Berlin Heidelberg, Berlin, Heidelberg (2004) 121–130.

[44] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller and E. Teller, Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21** (1953) 1087–1092.

[45] T.A. Gardner, M. Benzie, J. Börner, E. Dawkins, S. Fick, R. Garrett, J. Godar, A. Grimard, S. Lake, R.K. Larsen and N. Mardas, Transparency and sustainability in global commodity supply chains. *World Dev.* **121** (2019) 163–177.

[46] E. Adida and G. Perakis, A robust optimization approach to dynamic pricing and inventory control with no backorders. *Math. Program.* **107** (2006) 97–129.

[47] D. Alem, E. Curcio, P. Amorim and B. Almada-Lobo, A computational study of the general lot-sizing and scheduling model under demand uncertainty via robust and stochastic approaches. *Comput. Oper. Res.* **90** (2018) 125–141.