Journal of Industrial and Systems Engineering
Vol. 13, No. 4, pp. 62-80
Autumn (November) 2021

A two-level pricing-inventory-routing problem in green closed-loop supply chain: Bi-level programming and heuristic method

Mohsen Saffarian[1*], Sobhan Mostafayi[2], Seyed Mahmood Kazemi[1], Malihe Niksirat[3]

[1]*Department of Industrial Engineering, Birjand University of Technology, Birjand, Southern Khorasan, Iran*
[2]*Department of Industrial Engineering, University of Kurdistan, Sanandaj, Iran*
[3]*Department of Computer Science, Birjand University of Technology, Birjand, Southern Khorasan, Iran*
saffarian@birjandut.ac.ir, sobhan.mostafayi@gmail.com, kazemi_m_s@birjandut.ac.ir, niksirat@birjandut.ac.ir

Abstract

In this paper, a bi-level mathematical formulation for a pricing-inventory-routing problem in the context of sustainable closed-loop supply chains is developed. The two levels are entitled as the upper level model and the lower level model. The upper level model (the leader model) tries to minimize greenhouse gas (GHG) emissions while the lower level model (the follower model) focuses on profit maximization. To solve the problem, an enumeration heuristic method based on knapsack problem and genetic algorithm (GA) is devised. The results show that the heuristic method is capable of obtaining high-quality solutions in reasonable CPU-times.

Keywords: Bi-level programming, heuristic method, pricing-routing-inventory, closed loop supply chain, incentive loans

1- Introduction

Pricing is one of the main activities in economic topics, especially microeconomics (Christensen, 2013). Lack of a good pricing strategy has led many companies to supply goods or technologies with unfit prices for the market (Jain et al., 2003). Price is the most flexible element in marketing strategy and pricing decisions can be implemented relatively faster than other elements of marketing strategy (Avlonitis and Indounas, 2005). It is also one of the most challenging activities in the field of product commercialization, evaluation and pricing. Price adjustment is called pricing strategy. The goal of pricing strategy is setting optimal prices by maximizing current profits and the number of sales units (Dolgui and Proth, 2010). Although price competition is one of the major difficulties companies face, many of them have failed to establish a sound pricing strategy. It is despite the fact that pricing is one of the salient factors affecting customers' attraction, satisfaction and loyalty (Kotler and Armstrong, 2013). This study aims to provide a bi-level mathematical model for formulating a two-level supply chain where Department of Environment acts as leader and the chain owners as followers.

*Corresponding author

The decisions to be made include inventory management, pricing, routing, and adopting the appropriate decisions for collected returnable transport items (RTIs).
In a nutshell, the contributions of this paper are as follows.

- It is considered for the first time that in a closed-loop supply chain the leader is able to affect the decisions made by chain owners using allocation of financial incentives.

- A bi-level programming formulation is developed.

- Minimizing GHG emissions is considered as the leader objective and profit maximization as the follower objective.

- A new bi-level heuristic method based on knapsack problem and genetic algorithm (GA) is proposed.

This paper is organized as follows. In section 2 a brief literature review is provided. The developed formulation along with the employed heuristic approach are elaborated in sections 3, 4 and 5. Computational results and discussions are presented in section 6. Finally, section 7 covers all conclusions.

2- Literature review

In closed-loop supply chains, both forward and reverse activities are included (Govindan et al., 2015, Jangali, et al., 2020). One of the important problems in closed-loop supply chains is determining the appropriate Forward supply chain encompasses all activities from extracting raw materials to shipment of final products and backward flow starts from the point of consumption to the point of origin to find the best economic mode of reusing, reproducing and re-marketing for exploiting the new market (Khatami et al., 2015). Transition and collection routes along with the inventory levels at supply centers (Zhalechian et al., 2016). Closed-loop inventory-routing problem has been applied for optimization of various logistic systems including food supply chains (Forouzanfar et al., 2017). Choosing an appropriate pricing strategy is an important decision in logistics (Kaya and Urek, 2016). Since all parties across chain seek to maximize their own profits, an appropriate pricing strategy by considering various economic aspects can lead to creation of competitive advantages (Taleizadeh and Noori-daryan, 2016). Thus, integrating inventory-routing problem with product pricing problem can yield more practical solutions for managers and chain owners (Moghadas Poor, et al., 2021). In supply chains, the state, as a key decision-maker at the top of the chain, usually attempts to intervene in order to control the overall structure of the chain through applying some certain rules (Gao et al., 2016). It causes inconveniences in finding optimal solutions to the chain problems and conflicts between the leader and followers (chain owners) (Gao et al., 2016). It is a problem that has not been considered in majority of studies while chain owners are considered as the only decision-makers (Hsueh, 2015). Decision-making problems are often modeled as Stackelberg game in decentralized organizations and are formulated as bi-level programming problems (Munson and Rosenblatt, 2001). The leader determines high-level strategies and decisions based on which followers optimize their actions. By full awareness from the high-level decision maker's action, multi-level programming models use the concept of the equilibrium solution of Stackelberg (Wen and Hsu, 1991). Therefore, the leader must make a decision that optimizes their goal considering the follower's ensuing reaction (Gao et al., 2016). Roghanian et al. (2007) used two-level programming for modeling such decentralized decisions. Aviso et al. (2010) designed the chain of materials transportation to reduce and minimize resource consumption and waste production in eco-industrial parks, in order to encourage establishing an exchange network of produced wastes between plants that are located in eco-industrial parks. Figure 1 shows an example of the problem.

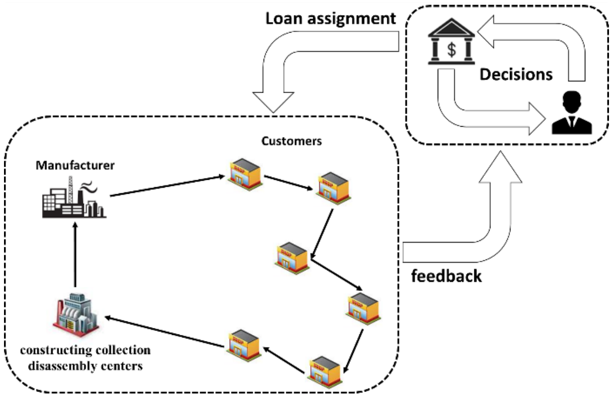

Fig 1. An example of the problem

Amirtaheri et al. (2017) provided a bi-level programming model considering production and distribution levels. In this model, price of products was also considered as a decision variable, and the equilibrium point of Stackelberg was achieved via genetic and particle swarm optimization (PSO) algorithms. Yue and You (2017) presented the problem of supply chain design in the form of a bi-level game and used a heuristic method to calculate the equilibrium point of Stackelberg.

In this work, the problem of pricing-inventory-routing optimization in closed-loop supply chain is studied and a multi-level programming model is developed. In our model, the state or Department of Environment aims to minimize the environmental effects of the chain by providing financial incentives and the chain owners pursue their profit maximization goals. Accordingly, in this situation, Department of Environment acts as a leader and owner of the supply chain is in the position of the follower (backward). The supply chain includes customers and collection and disassembly centers. The government (Department of Environment) is seeking to influence the chain by providing financial incentives for the centers as well as less-polluting vehicles. In the meantime, the follower problem is resolved based on a decision made by the leader, and the leader will ultimately make the best decision based on the follower's solutions.

Another application is the Vendor-Managed Inventory paradigm, when a vendor manages their own inventory in addition to those of its customers (Guimarães, et al., 2019). Therefore, a good platform can be created to make a competitive advantage in the chain by determining the price of the final product, considering all the economic considerations (Taleizadeh and Noori-daryan, 2016).

The application of the problem is applied in the supply chain programming throughout the organization.

Since, the existence of discrete variables at the follower level prevent us to use the KKT conditions, in this paper, the equilibrium point of Stackelberg is calculated using a heuristic algorithm. Golpîra, et al. (2017) formulated the problem of supply chain design in terms of uncertainty using bi-level programming. In this chain, the agility and lean production criteria are considered for the producer level and the KKT conditions are used to solve the problem. A mathematical formulation for a two-echelon inventory routing problem is proposed to minimize the overall cost in an integrated supply chain management (Farias, et al., 2019, Ebrahimi and Tavakkoli-Moghadam, 2020).

Considering the literature review and previous studies, there is not a research on the use of multi-level programming approach in closed loop supply chain design. However, considering actors as leader-follower levels can enhance the level of results for implementation in the real world (Rowshannahad, et al., 2018). Also, Competition between regular and closed loop supply chains was studied by a game theory approach (Hadi et al., 2021).

Table 1. Summery of researches on the two-echelon pricing-routing-inventory problem

Sustainability			Dimensions			Solution method				Type		Author(s)
Social	Environmental	Economic	Other	Routing	Location	Other	Metaheuristic	Heuristic	Exact	Bi-level	Single level	
✓	✓	✓		✓	✓		✓			✓		(Biuki, et al., 2020)
	✓	✓	✓		✓			✓		✓		(Kumar, et al., 2020)
		✓			✓		✓				✓	(Khalafi, et al. 2020)
		✓	✓		✓		✓				✓	(Babagolzadeh, et al., 2020)
	✓	✓	✓		✓	✓				✓		(Krishnan, et al., 2020)
		✓			✓		✓				✓	(Sherafati, et al., 2020)
		✓		✓	✓			✓			✓	(Imran, et al., 2020)
	✓	✓		✓	✓		✓			✓		(Gholizadeh, et al., 2020)
		✓			✓		✓				✓	(Liu, et al., 2020)
	✓		✓		✓			✓			✓	(Xu, et al., 2020)
	✓		✓		✓			✓			✓	(Forghani, et al., 2020)
	✓	✓	✓	✓	✓		✓			✓		This paper

Therefore, in this research, we investigate the problem of pricing-routing-inventory optimization in closed loop supply chain using multi-level programming in order to cover the existing research gap. In the provided model, the state or Department of Environment seeks to minimize the environmental effects of the chain by providing financial incentives and the chain owners are seeking to maximize their profitability. Therefore, in the defined problem, the state (Department of Environment) places in the position of leader (leading) and owner of the supply chain places in the position of follower (backward).

The considered supply chain includes customers and collection and disassembly centers. The government (Department of Environment) is seeking to influence the structure by providing financial incentives for the centers, as well as the usage of vehicles that produce less environmental damage. These financial incentives lead to provide the facilities and clean vehicles and reduce harmful environmental effects throughout the chain. In the meantime, the follower problem is resolved based on a decision made by the leader, and the leader will ultimately make the best decision based on the follower's answers. In the proposed solution approach, the leader level generated all its strategies based on a heuristic method and then the follower level generates its optimal answer to each strategy using genetic algorithm. Several numerical examples in different dimensions are generated to investigate the performance of the proposed model and algorithm. Table 1 summarizes the researches on the Two-echelon Pricing-Routing-Inventory Problem including the solution approach, dimensions and Sustainability. The table also demonstrates the contribution of the paper to the body of literature.

3- Problem statement

In this section, firstly, a bi-level programming formulation for the problem is developed. Noteworthy, the follower model is developed based on the model proposed by Mohammadnejad et al. (2016). Subsequently, the new bi-level heuristic approach is presented.

Assumptions of the problem are as follows:

1) Number of customers and their demands are known.
2) Return of products used by customers is specified.
3) A homogeneous fleet of vehicles is considered.
4) Costs of facilities location and product shipments are determined.
5) Selling price of products is specified.
6) A two-echelon closed loop supply chain is assumed.
7) The leader level includes government decisions and the follower level includes supply chain design.
8) All input parameters are deterministic.

Indexes and sets

M	Set of Products
N	Set of all nodes in the network
C	Set of Customers
L	set of potential places for constructing collection and disassembly centers
R	set of Raw materials
V	set of Vehicles
T	set of Programming periods

Parameters

γ_m^{Max}	Maximum defective rate of product m
γ_m^{Min}	Minimum defective product m
φ_r	A percentage of raw material r from defective products which can be reused
w_m	Volume percent of each unit of product m
FL_l	Cost of constructing the collection and disassembly center at a potential location l
CV_v	Vehicle capacity V
S_{mt}^{Max}	The maximum possible price for product m during period t
S_{mt}^{Min}	The minimum possible price for product m during period t
FV_{vt}	Cost of using the vehicle V during period t
DV_{vt}	Cost of travelled distance for vehicle V during period t for travelling a distance unit
SR_{rt}	Sale price of unusable raw material r in period t
α_{jmt}^{Max}	The maximum customer demand j for product m during period t
α_{jmt}^{Min}	The minimum customer demand j for product m during period t
h_{rt}	Maintenance cost of a raw material unit r in the manufacturer's warehouse
CR_{mr}	raw material consumption rate t for producing product m
A_{rt}	Ordering cost for raw material r during period t
PB_{rt}	Purchasing cost for raw material r during period t
$Dist_{ij}$	Distance between the two nodes i and j
LA	A big number
EX_{ijvt}	Rate of GHG emissions if vehicle v moves from customer i to customer j during period t
EY_l	Rate of GHG emission if collection and disassembly center is constructed at potential location l
EV_{vt}	Rate of GHG emissions if vehicle v is used during period t
α_l	The amount of loan (financial incentive) for constructing a collection and disassembly center at a potential location l
β_v	The amount of the loan (financial incentive) for using the vehicle V during period t
θ	The maximum loan amount for constructing a collection and disassembly center at a potential location l
ϑ	The maximum loan amount for using vehicle V during period t

Leader decision variables

IO_l	If a loan (financial incentive) is selected to construct collection and disassembly center at a potential location l, it is equal to one, otherwise it equals zero
IT_{vt}	If a loan (financial incentive) is selected to use vehicle V during period t, it is equal to one, otherwise it equals zero

Follower decision variable

LR_{vt}	Loading volume of vehicle V when departing from the manufacturer in period t
S_{mt}	Sales price of product m during period t
RW_{rt}	Number of raw materials r that are recycled during period t
Z_{jvt}	The variable related to the removal of sub tour of the vehicle V in the route of the node j during period t
D_{jmvt}	Number of products m delivered to customer J by vehicle V during period t
P_{jmvt}	Number of returned products m which are loaded from customer j by vehicle V during period t
LC_{jvt}	Load volume of vehicle V after exiting customer j during period t
I_{rt}	Inventory of raw material r in the producer's warehouse during period t
Q_{rt}	The amount of raw material r by the manufacturer during period t
X_{ijvt}	Is equal to one if the vehicle V moves from customer i to customer j during period t
Y_l	Is equal to one if the collection and disassembly center is constructed at the potential location l
O_{rt}	Is equal to one if the raw material r is ordered during period t
V_{vt}	Is equal to one if the vehicle V is used during period t

4- Mathematical formulation

Upper level (leader) model:

$$Minimizing = \sum_{i=1}\sum_{j=1}\sum_{v=1}\sum_{t=1} X_{ijvt}\ EX_{ijvt} + \sum_{l=1} Y_l EY_l + \sum_{v=1}\sum_{t=1} V_{vt}EV_{vt}$$

$$\sum_{l=1} IO_l \alpha_l \le \theta$$

$$\sum_{l=1} IT_{vt}\,\beta_v \le \vartheta$$

Lower Level (Follower) Model:

$$\begin{aligned} Maximizing\ & \sum_{t\in T}\sum_{v\in V}\sum_{m\in M}\sum_{j\in C} D_{jmvt}S_{mt} + \sum_{t\in T}\sum_{r\in R}\sum_{v\in V}\sum_{m\in M}\sum_{j\in C} SR_{rt}P_{jmvt}CR_{mr} \\ & - \sum_{t\in T}\sum_{v\in V}\sum_{m\in M}\sum_{j\in C} P_{jmvt}S_{mt} - \sum_{t\in T}\sum_{r\in R} O_{rt}A_{rt} - \sum_{t\in T}\sum_{r\in R} PB_{rt}O_{rt} - \sum_{t\in T}\sum_{r\in R} I_{rt}h_{rt} \\ & - \sum_{t\in T}\sum_{v\in V} V_{vt}FV_{vt} - \sum_{t\in T}\sum_{v\in V}\sum_{i\in N}\sum_{j\in N} X_{ijvt}DV_{vt}Dist_{ij} - \sum_{l\in L} Y_l FL_l + \sum_{l=1} IO_l\alpha_l Y_l \\ & + \sum_{l=1} IT_{vt}\,\beta_{vt}V_{vt} \end{aligned} \qquad 1$$

$s.t$

$$\sum_{v\in V} D_{jmvt} \le 1 + \alpha_{jmt}^{Max} - \frac{\alpha_{jmt}^{Max} - \alpha_{jmt}^{Min}}{S_{mt}^{Max} - S_{mt}^{Min}}\left(S_{mt} - S_{mt}^{Min}\right) \qquad j\in C, m\in M, t\in T \qquad 2$$

$$\sum_{v\in V} D_{jmvt} \ge \alpha_{jmt}^{Max} - \frac{\alpha_{jmt}^{Max} - \alpha_{jmt}^{Min}}{S_{mt}^{Max} - S_{mt}^{Min}}\left(S_{mt} - S_{mt}^{Min}\right) \qquad j\in C, m\in M, t\in T \qquad 3$$

$$I_{rt} = I_{rt-1} + Q_{rt} + RW_{rt} - \sum_{v\in V}\sum_{m\in M}\sum_{j\in C} D_{jmvt}CR_{mr} \qquad r\in R, t\in T \qquad 4$$

$$Q_{rt} \le O_{rt}\ LA \qquad r\in R, t\in T \qquad 5$$

$$\sum_{v\in V} P_{jmvt} \le 1 + \gamma_m^{Max} - \frac{\gamma_m^{Max} - \gamma_m^{Min}}{S_{mt}^{Max} - S_{mt}^{Min}}\left(S_{mt} - S_{mt}^{Min}\right)\sum_{v\in V} D_{jmvt-1} \qquad j\in C, m\in M, t\in T \qquad 6$$

$$\sum_{v\in V} P_{jmvt} \ge \gamma_m^{Max} - \frac{\gamma_m^{Max} - \gamma_m^{Min}}{S_{mt}^{Max} - S_{mt}^{Min}}\left(S_{mt} - S_{mt}^{Min}\right)\sum_{v\in V} D_{jmvt-1} \qquad j\in C, m\in M, t\in T \qquad 7$$

$$\sum_{v\in V}\sum_{i\in\{1,C\}} X_{ijvt} = 1 \qquad j\in C, t\in T \qquad 8$$

$$\sum_{i\in\{1,C\}} X_{ijvt} = \sum_{i\in\{C,L\}} X_{jivt} \qquad j\in C, v\in V, t\in T \qquad 9$$

$$\sum_{i\in C} X_{1ivt} \leq V_{vt} \qquad v \in V, t \in T \qquad 10$$
$$\sum_{l\in L} Y_l = 1 \qquad 11$$
$$\sum_{i\in C} X_{i1vt} \leq Y_l\, LA \qquad v \in V, l \in L, t \in T \qquad 12$$
$$V_{vt} \leq \frac{\sum_{i\in C} X_{1ivt} + \sum_{i\in C}\sum_{j\in C} X_{ijvt} + \sum_{i\in C} X_{i1vt}}{3} \qquad v \in V, t \in T \qquad 13$$
$$Z_{jvt} > Z_{ivt} - \left(1 - X_{ijvt}\right)LA \qquad (i,j) \in N, v \in V, t \in T \qquad 14$$
$$RW_{rt} < 1 + \varphi_r \sum_{v\in V}\sum_{m\in M}\sum_{j\in C} P_{jmvt} CR_{mr} \qquad r \in R, t \in T \qquad 15$$
$$RW_{rt} > \varphi_r \sum_{v\in V}\sum_{m\in M}\sum_{j\in C} P_{jmvt} CR_{mr} \qquad r \in R, t \in T \qquad 16$$
$$LR_{vt} = \sum_{m\in M}\sum_{j\in C} D_{jmvt}\, w_m \qquad v \in V, t \in T \qquad 17$$
$$LC_{ivt} \geq LR_{vt} - \sum_{m\in M} D_{imvt}\, w_m + \sum_{m\in M} P_{imvt}\, w_m - (1 - X_{1ivt})LA \qquad i \in C, v \in V, t \in T \qquad 18$$
$$LC_{jvt} \geq LC_{ivt} - \sum_{m\in M} D_{jmvt}\, w_m + \sum_{m\in M} P_{jmvt}\, w_m - \left(1 - X_{ijvt}\right)LA \qquad (i,j) \in C, v \in V, t \in T \qquad 19$$
$$LR_{vt} \leq CV_v V_{vt} \qquad v \in V, t \in T \qquad 20$$
$$LC_{ivt} \leq CV_v V_{vt} \qquad i \in C, v \in V, t \in T \qquad 21$$
$$S_{mt}^{Min} \leq S_{mt} \leq S_{mt}^{Max} \qquad m \in M, t \in T \qquad 22$$
$$X_{ijvt}, Y_l, Z_{jvt}, O_{rt}, V_{vt} \in \{0,1\} \qquad (i,j) \in C, v \in V, t \in T \qquad 23$$
$$D_{jmvt}, P_{jmvt}, RW_{rt} \in Integer \qquad j \in C, m \in M, v \in V\ t \in T \qquad 24$$

The objective function of the leader problem minimizes greenhouse gas emissions and the corresponding constraints shows the maximum available budget for funding financial incentives. In contrast, the follower problem focuses on profit maximization and its objective function consists of several parts. The first and second parts compute the revenue of selling products and wastes. The subsequent parts calculate system costs including delivery costs of defective products to customers, ordering costs, purchasing costs of raw materials, holding costs, vehicles usage costs, traveling costs and constructing costs of collection and disassembly centers. Variables IO_l and IT_{vt} in the last two parts of the objective function obtain their values from solving the leader model, i.e., considered as parameters to the follower model. Constraints (2) and (3) indicates the relation between demand of each product and its price for each customer in each period. Constraint (4) shows the inventory of raw materials in each period in the producer's warehouse. Constraint (5) denotes that prior to receiving raw materials and order must be established. Constraints (6) and (7) calculate the amount of returned products from each customer in each period. Constraint (8) ensures that all customers must be serviced. Constraint (9) ensures that any vehicle that enters a node must leave it. Constraint (10) shows when a vehicle can be used Constraint (11) ensures that one of the potential points should be chosen to build collection and disassembly centers. Constraint (12) guarantees that vehicles do not move from constructed centers in order to service. A movement must start from and finish to the manufacturer which is enforced by constraint (13). Subtour elimination constraints are imposed by constraint (14). Constraints (15) and (16) calculate the number of reusable products collected from customers. Constraint (17) calculates load volume of vehicles when departing from the manufacturer. Constraint (18) calculates vehicle's load volume after the first customer's exit, and the constraint (19) shows vehicle's load volume after visiting each customer. Constraint (20) shows vehicle's load volume after withdrawal from the manufacturer. Constraint (21) ensures that vehicle's amount of load does not exceed its capacity. Constraint (22) determines price interval of each product in each period. Integrality and non-negativity conditions on variables are imposed by constraints (23) and (24).

5- The proposed algorithm

Due to presence of binary variables in the follower model, the proposed formulation cannot be converted to a single-level one. Thus, we focused on the classic Stackelberg game, which has a leader and a follower. The leader here seeks to minimize his goal based on which the follower optimizes their objective function. Note that the leader's decision is considered as an input to the follower model. It is clear that solution space of the follower model is affected by the leader's decisions.

Variables of the leader model determine financial incentive allocations. As a result, an enumeration method is devised to count all possible allocations. Based on each allocation type, the follower model is solved and the obtained solution is analyzed. This method is suitable for small problem instances. Based on the idea mentioned here, a heuristic method is developed which is delineated in the subsequent parts.

The heuristic algorithm proposed in this paper is a bi-level algorithm that integrates the enumeration method explained before with the logic of simulated annealing (SA) algorithm. To solve and make various decisions by the leader, a fake knapsack problem is used to examine the different modes of allocating financial incentives. After solving the knapsack problem with the value of C different coefficient in each repetition, the problem's solutions are inserted in the follower problem and the follower problem is solved once for each solution. After storing the solutions of the follower problems, the value of its variables is placed in each replication in the leader's objective function, and the minimum amount of the leader problem among all solutions is considered as the optimal solution. The fake knapsack used is as follows:

Upper level (leader) model:

$$Maximizing = \sum_{l=1} IO_l + \sum_{v=1} IT_v$$

$$\sum_{l=1} R_l \mathrm{IO_l} \leq \theta$$

$$\sum_{l=1} E_{vt}\, IT_v \leq \vartheta$$

Where the parameters R_l and E_{vt} are randomly generated numbers (fake weights) used to create different strategies for the leader. Step-by-step description of the algorithm is as follows.

Step 1: The values of financial incentive allocation variables are firstly determined by the fake knapsack problem. Afterwards, the obtained values are used for solving the follower model.

Step 2: Solve the follower model formulation and store the obtained optimal solution.

Step 3: Create neighbor solutions using the following equation

R_ New=R_ Initialized+α.R

E_ New=E_ Initialized+α.E

Where parameters R and E have random structures explained in the following, R_ New and E_ New are updated values of R and E, R_ Initialized and E_ Initialized are the initial produced values of the parameters R and E and α is a coefficient for producing different solutions.

Step 4: Considering the follower problem's solutions, the value of the objective function of the leader problem is calculated in each repetition using the logic of SA algorithm while bad solutions can also be accepted with a predetermined probability.

Step 5: The optimal value of the objective function of the leader problem equals to the minimum value in all repetitions .

Step 6: Repeat Steps 1 through 5 until the stopping criteria is satisfied.

The following pseudo code shows structure of the proposed algorithm:

Step 1. Generating various decision strategies of the leader level
Step 2. Storing values of the leader-level decision variables for each strategy generated in Step 1 in set *Strategies*
Step 3. Solving the follower problem for the values of the leader-level variables stored in set *Strategies*
Step 4. Storing the values obtained from Step 3 in set *Initial Solutions*
Step 5. Placing each of the decision variables in the Strategies and Initial Solutions sets in the leader model and calculating (without optimization) the leader model for each member of the sets
Step 6. Selecting the best solution based on the values obtained in Step 5 as the final solution

Considering that the follower problem is NP-hard, it is advisable to use metaheuristic algorithms to solve it. Consequently, in this study, a genetic algorithm based on the structure proposed by Mohammadnejad et al. (2016) is developed. The aim is using the algorithm to create semi-feasible solutions in problem-solving process. The set of optimal solutions of the follower model form the leader's solution space and the leader is to obtain the most appropriate solution in this space. If the solutions produced by the follower are not optimal, the solution space for the leader will not be spatially optimal and close-to-optimal points will be formed. Accordingly, the final solution of the leader will also be semi-feasible, which can be regarded as valid solutions (Yue & You, 2017).

Pseudo code of the GA is presented below.

***Input**: fitness function, max iteration, Population size, Crossover rate, Mutation rate*
***Output**: the elitist*
Initialize a population randomly
Calculate the fitness of the population and find the elite
  $t = 0$
  While** $t \leq T$ **do
    Perform crossover using a two-point crossover operator
    Perform Mutation
    Carry out the replacement strategy and evaluate the solutions
    Calculate the fitness and return the elite population
    $t = t + 1$
  End
  Final solution ← the elite
End
***Return** Final Solutions*

6- Computational results

In this section, behavior of the proposed method is investigated using a small-sized test problem extracted from Zeballos et al. (2014). In this test problem there are 10 customers serviced by three potential collection and disassembly centers. The cost of establishing each center is described in table 2.

Table 2. The cost of building each center

Potential collection and disassembly center	1	2	3
Cost	670000	860000	700000

There are also vehicles 1 and 2 with capacities of 17000 tons and 15000 tons, respectively. Three raw materials are used with volume coefficients of 2, 1 and 1, that the usable material rate is 0.71%, 0.74% and 0.74%, respectively. Also, there are two final products with volume coefficients of 9 and 6, respectively. Raw material consumption coefficient in product 1 equals to 1, 1, 3, and 3, 1, 3 in product 2, respectively. At the leader level, the parameters of the objective function in kilograms of carbon dioxide are in the following ranges $EX_{ijvt} = [2,3]$ and $EY_l = [50,80]$ and$EV_{vt} = [3,5]$. The values of incentive loans for each center is shown in table 3.

Table 3. Leader level parameters

The amount of incentive loan for each of the potential centers for constructing collection and disassembly centers					**The amount of incentive loans for the use of vehicles**		
Potential collection and disassembly center	1	2	3		Vehicle	1	2
Cost	60000	80000	100000		Cost	1000	800

Since it is a small-sized test problem, the follower model can be solved by Cplex solver. As explained before, the leader considers a number of lending strategies for generating initial solutions. These strategies are presented in Table 3. The maximum amount of loans for constructing potential centers is 200,000 and for vehicles is 2000 and the total number of selectable strategies for the leader equals to 6 for constructing centers and 3 for lending vehicles. Therefore, there are totally 6 × 3 = 18 strategies

Table 4. Different lending strategies for constructing centers and purchasing vehicles

Strategy	**Construction Loan Center**	**Vehicle loan**
1	Center 1	Device 1
2	Center 1	Device 1
3	Center 1	Device 1&2
4	Center 2	Device 1
5	Center 2	Device 2
6	Center 2	Device 1&2
7	Center 2	Device 1
8	Center 2	Device 2
9	Center 2	Device 1&2
10	Center 1&2	Device 1
11	Center 1&2	Device 2
12	Center 1&2	Device 1&2
13	Center 1&2	Device 1
14	Center 1&2	Device 2
15	Center 1&2	Device 1&2
16	Center 1&2	Device 1
17	Center 1&2	device
18	Center 1&2	Device 1&2

Therefore, it is sufficient to initiate solving of the follower problem for all the strategies mentioned above where leader-level variables including IO_l and IT_v are parameters to the follower-model; this is done by Cplex Solver. Then the set of generated solutions is considered as the leader's decision-making space and this problem is solved again by Cplex Solver. After solving the leader problem, it is obvious that the selected solution is considered as the final solution for the problem. After implementing the described process, the generated solutions in each repetition are presented in figure 2.

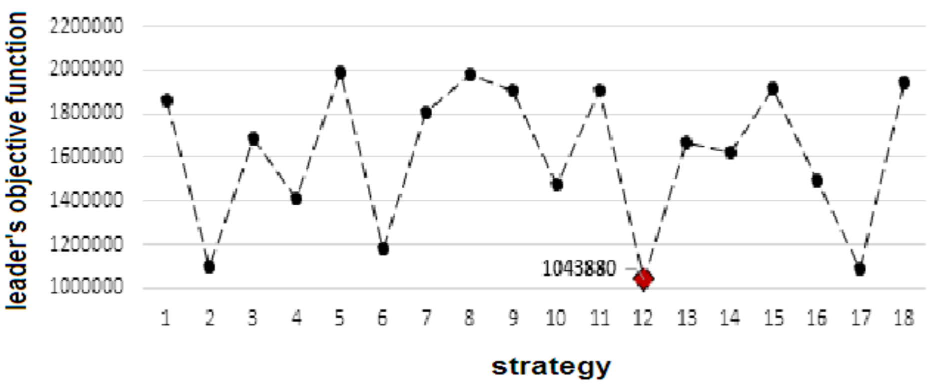

Fig 2. The leader's objective function for the optimal solutions of the manufacturing strategies

As can be seen, strategy 12 has the least objective function value and is selected as the final solution. An important point seen in figure 2 is the significant difference between the values of the leader's objective function for different strategies. This is due to the large differences in the generated solutions from solving the follower problem for different strategies. But in the final selected solution with the leader's objective function value of 1043880, the follower problem structure is as follows. The collection center 2 is selected and used to cover the customers by vehicle 2 in period 1. In period 2 both vehicles 1&2 are used, allocating customers 1, 3, 5, 6, 9 and 10 to vehicle 1 and customers 2, 4, 7 and 8 to vehicle 2. The values for different parts of the objective function are given in table 5.

Table 5. The Follower level costs for the best strategy (strategy 12)

objective function	Ordering cost	purchasing	Maintaining	Applying vehicles	Routing	Constructing centers
75450843	860000	1051447	520000	2180054	67493584	3345758

6-1- Test problems

Before explaining the final solutions, it is necessary to examine the efficiency of the proposed GA. To do so, a number of numerical problem instances are designed and the results are given in table 5. To illustrate the efficiency of the provided algorithm, we use the deviation percentage indicator from the best solution for each test problem. The general formula for calculating this index is shown in the following equation:

$$RPD = \frac{f(s)-f(s^*)}{f(s^*)} \times 100$$

In this formula, $f(s)$ represents the value obtained by the discussed algorithm and mathematical model, while $f(s^*)$ represents the best value ever obtained for that problem. This indicator can well represent efficiency of the algorithm (Soleimani, 2015).

Table 6. Evaluating the results of the Cplex solver and the genetic algorithm

	Mathematical model		***Solving algorithm***			
Set	***RPD***	***time***	***RPD***			***time***
			Best	***Average***	***Worst***	
1	0	237.251	-0.0128	0.0529	0.1147	16.489
2	0	251.361	0	0	0	15.623
3	0	248.306	0	0.0514	0.1277	16.637
4	0	136.267	0.0765	0.0765	0.0765	14.266
5	0	417.242	0	0.0203	0.1012	14.251
6	0	157.925	0	0	0	16.331
7	0	125.481	0	0	0	14.288
8	0	137.224	0.1517	0.1517	0.1517	16.336
9	0	99.042	0	0.0208	0.0254	16.449
10	-	> 3 hrs.	0	0.1266	0.2417	37.411
11	-	> 3 hrs.	0	0.1222	0.2651	37.447
12	-	> 3 hrs.	0	0.1235	0.3366	38.417
13	-	> 3 hrs.	0	0.2323	0.4278	39.469
14	-	> 3 hrs.	0	0.2111	0.4808	37.260
15	-	> 3 hrs.	0	0.174	0.5728	37.296
16	-	> 3 hrs.	0	0.3939	0.7627	37.355
17	-	> 3 hrs.	0	0.2112	0.5398	37.581
18	-	> 3 hrs.	0	0.2351	0.9887	37.831
19	-	> 3 hrs.	0	0.2123	0.3737	620.470
20	-	> 3 hrs.	0	0.0834	0.3574	615.157

According to the information provided in table 5, it can be seen that the GA algorithm is capable of producing high-quality solutions in term of RPD indicator. In order to investigate the performance of the proposed heuristic algorithm, several test problems are produced and the obtained results are analyzed. For this purpose, according to the work of Soleimani and Kannan (2015), three different categories for small-, medium- and large-sized tests problems are considered each of which contains five test problems resulting in a total of 15 problems. In these problems, parameters are randomly generated in the specified intervals. Also, network nodes are provided in a random square space with side length of 200 units. Distances between these nodes are calculated based on Euclidean distance. Table 6 depicts the results obtained by the heuristic algorithm for each test problem in each category.

Table 7. Computational result for different instances

	Test problem	Run **1**	Run **2**	Run 3	Run 4	Run 5	Standard deviation
	1	1172771	1151713	1101225	1025616	1078297	52535/43666
	2	1168496	1016875	1150272	1178475	1001186	77465/38758
small	3	1075076	1072871	1029720	1174413	1018726	54956/3582
	4	1042600	1172185	1143327	1064266	1037049	55304/93042
	5	1007639	1165655	1139815	1047626	1194884	71489/41498
	6	1691912	1689636	1558811	1611090	1690307	54347/23839
	7	1517511	1695415	1561688	1581662	1639590	62091/06062
medium	8	1575114	1525999	1599553	1557461	1514904	31124/17437
	9	1679330	1572531	1622676	1660114	1691231	43114/37953
	10	1535953	1553607	1587495	1668536	1593470	45624/15048
	11	2476281	2533566	2523732	2423597	2454395	41486/79782
	12	2582708	2530262	2485152	2565570	2551059	33644/35463
large	13	2597641	2443996	2522645	2426908	2496461	60839/91181
	14	2422879	2541612	2574172	2573870	2507940	56209/90041
	15	2459765	2540396	2586398	2501571	2517775	41929/82544

According to table 6, in various test problems variations of solutions is acceptable which is a criterion for algorithm efficiency. As mentioned previously, the quality of generated solutions is largely dependent on the solutions generated by the follower-level model. Since the performance of the GA was shown to be at a convincing level, the space for leader's decisions is also satisfying. Due to the existing randomness in producing different strategies at the leader level, it is possible to confront with similar solutions. Figure 3 illustrates the objective functions of the solutions generated in the second stage of the proposed algorithm.

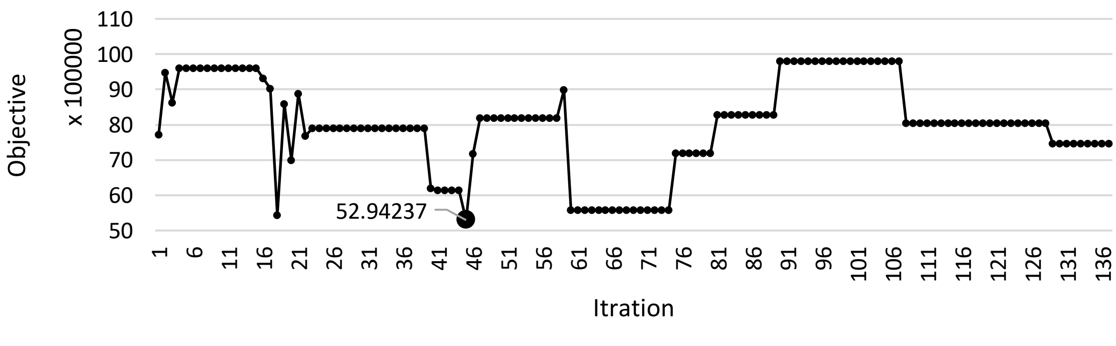

Fig 3. Different generated answers from problem solving using the heuristic algorithm

Figure 3 indicates that the algorithm produced the same solution levels in a number of iterations. Indeed, these solutions generated by the GA are equal to the specified strategies for solving the follower problem creating an acceptable space for the leader's problem. By changing the strategies, the generated solutions change but since the algorithm is not evolutionary, the objective function values of the solutions found in different iterations does not conform to a descending pattern. Therefore, the solution with the minimum objective function value is desirable which the last solution is not necessarily.

6-2- Performance evaluation of the bi-level algorithm

In this section, by solving a variety of numerical samples, the numerical results of solving 30 random test problems are investigated. It is noteworthy that each of the problems is performed 5 times independently and the results are recorded 5 times due to the random nature of the proposed method. These results are employed for producing strategies. Table 8 illustrates the characteristics of the small, medium and large problems. In tables 9 – 11, the optimal values of the follower objective function are presented for small, medium and large problems, respectively.

Table 8. Characteristics of the small, medium and large problems

Problem No.	Number of capacity levels	Number of vehicles	Number of collection centers	Number of customers	Size
1	3	2	3	15	Small
2	3	2	4	20	
3	3	2	7	20	
4	4	3	7	22	
5	4	2	5	25	
6	4	3	9	30	
7	5	4	10	30	
8	5	4	10	32	
9	7	5	12	35	
10	8	5	15	35	
1	10	7	10	50	Medium
2	10	8	12	50	
3	12	8	15	55	
4	15	9	18	55	
5	17	10	20	60	
6	17	10	25	60	
7	18	10	27	60	
8	20	12	30	65	
9	20	12	32	65	
10	20	15	35	65	
1	12	10	15	80	Large
2	15	10	15	90	
3	15	12	17	100	
4	18	15	20	100	
5	18	15	20	100	
6	20	15	25	100	
7	20	15	25	100	
8	20	15	25	110	
9	20	20	30	110	
10	25	20	35	110	

Based on the test problems presented in Table 8, the following results are obtained.

Table 9. The results of the small problems (* 1000)

Problem No.	Run 1	Run 2	Run 3	Run 4	Run 5	Best	Average	Worst
1	22927	23100	48753	29655	31691	22927	31225.2	48753
2	23386	23562	51191	30841	32959	23386	32387.61	51191
3	24555	24033	53238	32383	33288	24033	33499.56	53238
4	25291	24995	55900	33031	33954	24995	34634.23	55900
5	25544	26244	57577	33361	34973	25544	35539.94	57577
6	25800	26507	59305	33695	36022	25800	36265.54	59305
7	27090	27832	62270	34369	36382	27090	37588.48	62270
8	27903	28389	64761	35056	36746	27903	38570.74	64761
9	29019	29524	65408	36809	37113	29019	39574.64	65408
10	30179	30410	68679	38281	38969	30179	41303.62	68679

Table 10. The results of the medium problems (* 1000)

Problem No.	Run 1	Run 2	Run 3	Run 4	Run 5	Best	Average	Worst
1	26137	25410	55091	33807	34860	25410	35060.89	55091
2	27203	26648	58869	35497	36584	26648	36960.28	58869
3	26765	26210	61224	36883	37652	26210	37746.75	61224
4	28857	27247	62512	36337	38123	27247	38615.18	62512
5	29986	29376	61525	39889	41084	29376	40372.01	61525
6	30091	29670	62185	40763	42648	29670	41071.30	62185
7	30414	29765	66921	40106	40452	29765	41531.41	66921
8	32227	30062	65885	42884	42437	30062	42699.14	65885
9	33531	32145	69084	42142	43332	32145	44046.85	69084
10	35192	32840	73180	47118	48027	32840	47271.21	73180

Table 11. The results of the large problems (* 1000)

Problem No.	Run 1	Run 2	Run 3	Run 4	Run 5	Best	Average	Worst
1	28228	29222	62253	36511	39043	28228	39051.30	62253
2	30467	30112	66522	38337	40608	30112	41209.41	66522
3	29709	29618	67346	40940	42547	29618	42031.89	67346
4	32608	29699	69389	40335	43079	29699	43021.78	69389
5	32984	32314	68908	45074	46014	32314	45058.98	68908
6	32799	33231	69026	45654	47765	32799	45694.94	69026
7	34063	33634	72943	44117	44497	33634	45850.96	72943
8	35772	33069	75767	48459	46257	33069	47864.79	75767
9	37890	34716	75302	47621	46799	34716	48465.50	75302
10	38711	36452	83425	52772	54750	36452	53222.14	83425

As can be seen, the best, average and worst values for different implementations are fairly close, implying a satisfactory performance because of small deviations. Figures 4, 5 and 6 show the difference between the average, worst and best solution values in different runs.

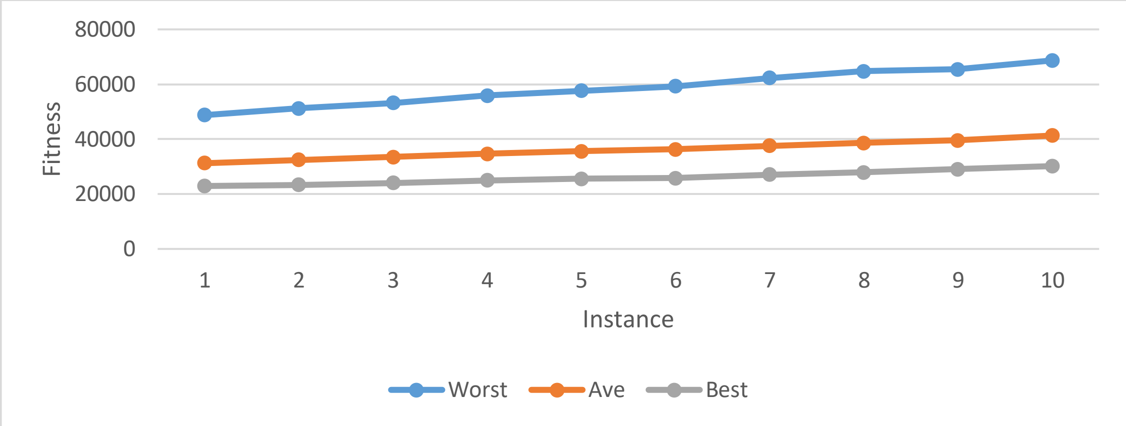

Fig 4. Comparison between the averages, worst and best solution values for small problems

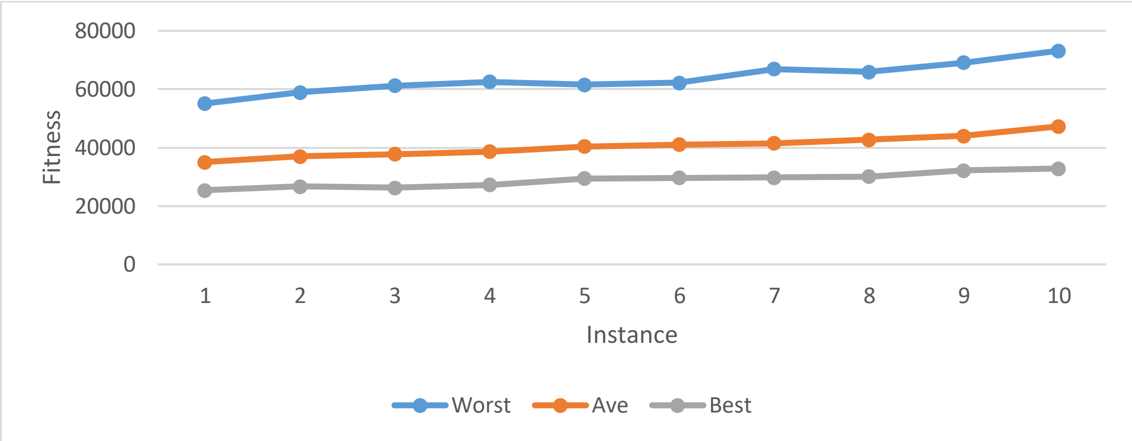

Fig 5. Comparison between the averages, worst and best solution values for medium problems

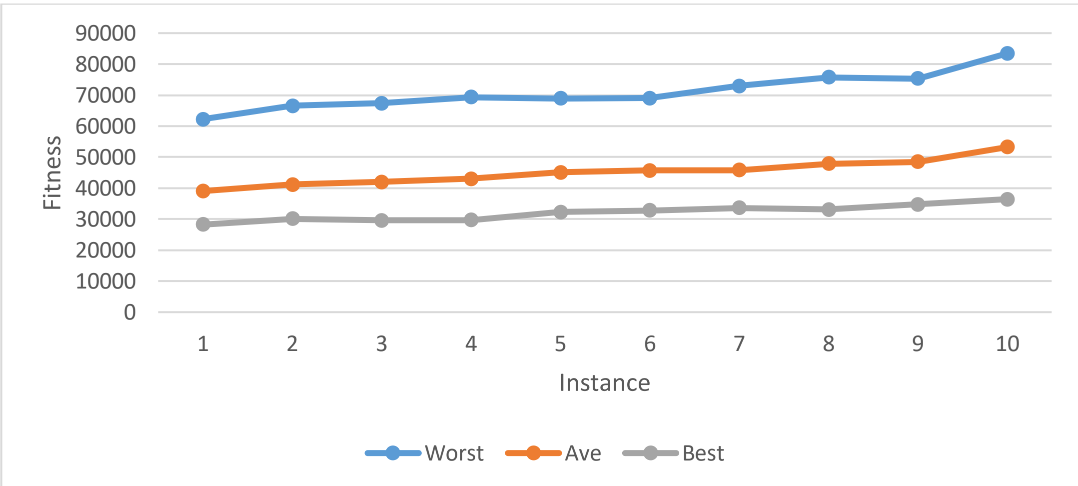

Fig 6. Comparison between the averages, worst and best solution values for large problems

7- Conclusion

In this study, a bi-level programming formulation for a pricing-inventory-routing problem in a bi-level closed-loop supply chain was proposed. The provided model included two sub-models; the leader model and the follower model. The main focus of the leader model was on minimizing GHG emissions while the follower model had a profit maximization objective. The main idea behind solving the problem was to solve the lower-level model and replacing the obtained variable values in the higher-level model. Compared to single-level programming models, the bi-level programming formulation had two advantages. Firstly, the bi-level programming was able to analyze two different and incompatible objective functions at the same time. Secondly, multi-criteria decision making method of bi-level programming could better reflect the real problem. To solve the developed formulation a bi-level heuristic algorithm was devised. It was considered for the first time that in a closed-loop supply chain the leader was able to affect the decisions made by chain owners using allocation of financial incentives. Since the follower model was NP-hard, a GA was embodied into the proposed heuristic approach. Using some data sets taken from the literature in three categories of small, medium and large sizes the performance of the GA and the proposed heuristic method were evaluated. The obtained results showed that both algorithms are capable of producing high-quality solutions in satisfactory CPU times .

Considering uncertainties in the problem and developing exact solution methodologies capable of solving larger problem instances to optimality can be regarded as future research opportunities.

References

Amirtaheri, O., Zandieh, M., Dorri, B., & Motameni, A. (2017). A bi-level programming approach for production-distribution supply chain problem. Computers & Industrial Engineering, 110, 527-537.

Aviso, K. B., Tan, R. R., Culaba, A. B., & Cruz, J. B. (2010). Bi-level fuzzy optimization approach for water exchange in eco-industrial parks. Process Safety and Environmental Protection, 88(1), 31-40.

Avlonitis, G. J., & Indounas, K. A. (2005). Pricing objectives and pricing methods in the services sector. Journal of services marketing, 19(1), 47-57.

Babagolzadeh, M., Shrestha, A., Abbasi, B., Zhang, Y., Woodhead, A., & Zhang, A. (2020). Sustainable cold supply chain management under demand uncertainty and carbon tax regulation. Transportation Research Part D: Transport and Environment, 80, 102245.

Biuki, M., Kazemi, A., & Alinezhad, A. (2020). An integrated location-routing-inventory model for sustainable design of a perishable products supply chain network. Journal of cleaner production, 120842.

Christensen, C. (2013). The innovator's dilemma: when new technologies cause great firms to fail: Harvard Business Review Press.

Dolgui, A., & Proth, J.-M. (2010). Pricing strategies and models. Annual Reviews in Control, 34(1), 101-110.

Ebrahimi M. and Tavakkoli-Moghadam R. (2020). Benders decomposition algorithm for a green closed-loop supply chain under a build-to-order environment, Journal of Industrial and Systems Engineering, 13, 102-111.

Farias, K., Hadj-Hamou, K., & Yugma, C. (2019). Mathematical formulations for a two-echelon inventory routing problem. IFAC-PapersOnLine, 52(13), 1996-2001.

Forghani, A., Dehghanian, F., Salari, M., & Ghiami, Y. (2020). A bi-level model and solution methods for partial interdiction problem on capacitated hierarchical facilities. Computers & Operations Research, 114, 104831.

Forouzanfar, F., Tavakkoli-Moghaddam, R., Bashiri, M., Baboli, A., & Molana, S. H. (2017). New mathematical modeling for a location–routing–inventory problem in a multi-period closed loop supply chain in a car industry. Journal of Industrial Engineering International, 1-17.

Gao, J., Han, H., Hou, L., & Wang, H. (2016). Pricing and effort decisions in a closed loop supply chain under different channel power structures. Journal of Cleaner Production, 112, 2043-2057.

Gholizadeh, H., Fazlollahtabar, H., & Khalilzadeh, M. (2020). A robust fuzzy stochastic programming for sustainable procurement and logistics under hybrid uncertainty using big data. Journal of cleaner production, 120640.

Golpîra, H., Najafi, E., Zandieh, M., & Sadi-Nezhad, S. (2017). Robust bi-level optimization for green opportunistic supply chain network design problem against uncertainty and environmental risk. Computers & Industrial Engineering, 107, 301-312.

Govindan, K., Soleimani, H., & Kannan, D. (2015). Reverse logistics and closed loop supply chain: A comprehensive review to explore the future. European journal of operational research, 240(3), 603-626.

Guimarães, T. A., Coelho, L. C., Schenekemberg, C. M., & Scarpin, C. T. (2019). The two-echelon multi-depot inventory-routing problem. Computers & Operations Research, 101, 220-233.

Hsueh, C.-F. (2015). A bilevel programming model for corporate social responsibility collaboration in sustainable supply chain management. Transportation Research Part E: Logistics and Transportation Review, 73, 84-95.

Hadi, T., Sheikhmohammady, M., Chaharsooghi, S. K., & Hafezalkotob, A. (2021). Competition between regular and closed loop supply chains under financial intervention of government; a game theory approach. Journal of Industrial and Systems Engineering, 13(2), 179-199.

Imran, M., Salman Habib, M., Hussain, A., Ahmed, N., & M Al-Ahmari, A. (2020). Inventory Routing Problem in Supply Chain of Perishable Products under Cost Uncertainty. Mathematics, 8(3), 382.

Jain, R. K., Martyniuk, A. O., Harris, M. M., Niemann, R. E., & Woldmann, K. (2003). Evaluating the commercial potential of emerging technologies. International journal of technology transfer and commercialisation, 2(1), 32-50.

Jangali, M. T., Makui, A., & Dehghani, E. (2020). Designing a closed loop supply chain network for engine oil in an uncertain environment: A case study in Behran Oil Company. Journal of Industrial and Systems Engineering, 13(2), 49-64.

Kaya, O., & Urek, B. (2016). A mixed integer nonlinear programming model and heuristic solutions for location, inventory and pricing decisions in a closed loop supply chain. Computers & Operations Research, 65, 93-103.

Khalafi, S., Hafezalkotob, A., Mohamaditabar, D., & Sayadi, M. K. (2020). A novel model for a network of a closed loop supply chain with recycling of returned perishable goods: A case study of dairy industry. Journal of Industrial and Systems Engineering, 12(4), 136-153.

Khatami, M., Mahootchi, M., & Farahani, R. Z. (2015). Benders' decomposition for concurrent redesign of forward and closed loop supply chain network with demand and return uncertainties. Transportation Research Part E: Logistics and Transportation Review, 79, 1-21.

Kotler, P., & Armstrong, G. (2013). Principles of Marketing (16th Global Edition): Harlow: Pearson.

Krishnan, R., Agarwal, R., Bajada, C., & Arshinder, K. (2020). Redesigning a food supply chain for environmental sustainability–An analysis of resource use and recovery. Journal of cleaner production, 242, 118374.

Kumar, A., Mangla, S. K., Kumar, P., & Karamperidis, S. (2020). Challenges in perishable food supply chains for sustainability management: A developing economy perspective. Business Strategy and the Environment.

Liu, W., Wan, Z., Wan, Z., & Gong, B. (2020). Sustainable recycle network of heterogeneous pharmaceuticals with governmental subsidies and service-levels of third-party logistics by bi-level programming approach. Journal of cleaner production, 249, 119324.

Mohammadnejad, M., Abadi, I. N. K., Sadeghian, R., & Ahmadizar, F. (2016). An Efficient Imperialistic Competitive Algorithm for the Closed loop Supply Chains Considering Pricing tor Product, and Fleet of Heterogeneous Vehicles. Transylvanian Review.

Moghadas Poor, B., Jabalameli, M. S., & Bozorgi-Amiri, A. (2020). A closed loop supply chain network design with considering third party logistics: A case study. Journal of Industrial and Systems Engineering, 13(1), 92-110.

Munson, C. L., & Rosenblatt, M. J. (2001). Coordinating a three-level supply chain with quantity discounts. IIE Transactions, 33(5), 371-384.

Roghanian, E., Sadjadi, S. J., & Aryanezhad, M.-B. (2007). A probabilistic bi-level linear multi-objective programming problem to supply chain programming. Applied Mathematics and computation, 188(1), 786-800.

Rowshannahad, M., Absi, N., Dauzère-Pérès, S., & Cassini, B. (2018). Multi-item bi-level supply chain programming with multiple remanufacturing of reusable by-products. International Journal of Production Economics, 198, 25-37.

Sherafati, M., Bashiri, M., Tavakkoli-Moghaddam, R., & Pishvaee, M. S. (2020). Achieving sustainable development of supply chain by incorporating various carbon regulatory mechanisms. Transportation Research Part D: Transport and Environment, 81, 102253.

Soleimani, H., & Kannan, G. (2015). A hybrid particle swarm optimization and genetic algorithm for closed loop supply chain network design in large-scale networks. Applied Mathematical Modelling, 39(14), 3990-4012.

Taleizadeh, A. A., & Noori-daryan, M. (2016). Pricing, manufacturing and inventory policies for raw material in a three-level supply chain. International Journal of Systems Science, 47(4), 919-931.

Wen, U.-P., & Hsu, S.-T. (1991). Linear bi-level programming problems—a review. Journal of the operational research society, 42(2), 125-133.

Xu, J., Deng, Y., Shi, Y., & Huang, Y. (2020). A bi-level optimization approach for sustainable development and carbon emissions reduction towards construction materials industry: a case study from China. Sustainable cities and society, 53, 101828.

Yue, D., & You, F. (2017). Stackelberg-game-based modeling and optimization for supply chain design and operations: A mixed integer bilevel programming framework. Computers & Chemical Engineering, 102, 81-95.

Zeballos, L. J., Méndez, C. A., Barbosa-Povoa, A. P., & Novais, A. Q. (2014). Multi-period design and programming of closed loop supply chains with uncertain supply and demand. Computers & Chemical Engineering, 66, 151-164.

Zhalechian, M., Tavakkoli-Moghaddam, R., Zahiri, B., & Mohammadi, M. (2016). Sustainable design of a closed loop location-routing-inventory supply chain network under mixed uncertainty. Transportation Research Part E: Logistics and Transportation Review, 89, 182-214.