Industrial Engineering
& Management Systems
Vol 13, No 4, December 2014, pp.383-397
ISSN 1598-7248 | EISSN 2234-6473 |
http://dx.doi.org/10.7232/iems.2014.13.4.383

Reverse Logistics Network Design with Incentive-Dependent Return

Mohammad Asghari*
Department of Industrial Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

Salman J. Abrishami
Department of Industrial Engineering, University of Bojnourd, Bojnourd, Iran

Faezeh Mahdavi
Department of Computer Engineering, Mazandaran University of Science and Technology, Babol, Iran

(Received: May 19, 2014 / Revised: October 1, 2014 / Accepted: October 30, 2014)

ABSTRACT

Reverse logistics network design issues have been popularly discussed in recent years. However, few papers in the past literature have been dedicated to incentive effect on return quantity of used products. The purpose of this study is to formulate a dynamic nonlinear programming model of reverse logistics network design with the aim of managing the used products allocation by coordinating the collection centers and recovery facilities to warrant economic efficiency. In the optimization model, a fuzzy approach is applied to interpret the relationship between the rate of return and the suggested incentives. Due to funding constraints in setting up the collection centers, this work considers these centers as multi-capacity levels, which can be opened or closed at different periods. In view of the fact that the problem is known as NP-hard, we propose a heuristic method based on tabu search procedure to solve the presented model. Finally, several dominance properties of optimal solutions are demonstrated in comparison with the results of a state-of-the-art commercial solver.

Keywords: Reverse Logistics, Incentive-Dependent Return, Nonlinear Programming, Network Optimization, Heuristic Algorithm

* Corresponding Author, E-mail: mohammad.asghari@stu-mail.um.ac.ir

1. INTRODUCTION

In recent decades, various industrial sectors have been affected by the growth of environmental concerns, introduction of new regulations and the increase in the rate of return as a result of shortened product life cycle. Accordingly, with the intention of passing the obligatory regulations, the manufacturers have been compelled to retrieve and recover products that are at the end of their life cycle (see Ilgin and Gupta, 2010).

Planning for the collection of used products is related to the reverse logistics network design, which includes not only selecting appropriate paths and vehicles for transporting goods but also determining the optimal locations and capacities for opening the collection and recovery centers. In this area, there are some challenging decisions to be made when various products and varying levels of quality are concerned. This is why a wide variety of mathematical optimization models have been published to deal with different situations in network design, which usually considers different stages for reverse logistics networks.

The literature addresses the collection of returned products in three ways: third-party logistics (Cruz-Rivera and Ertel, 2009; de Figueiredo and Mayerle, 2008; Govindan *et al*., 2012; Krikke *et al*., 2008; Min and Ko, 2008), opening collection centers by the remanufacturer/ manufacturer (Aras and Aksen, 2008; Östlin *et al*., 2008; Pokharel and Liang, 2012; Tagaras and Zikopoulos, 2008), and the use of retailers (Choi *et al*., 2013; Lee *et al*., 2011; Wojanowski *et al*., 2007).

The majority of the used products do not have any value in terms of functionality (Schultmann *et al*., 2006); only a reasonable amount of profit is made when some of them are collected and recycled. Imposing penalties for non-collected, end-of-life (EOL) products or pollutant limitations generated during the production process are samples of new approaches that have actively been pursued in the past decade. For example, in the Waste Electrical and Electronic Equipment (WEEE) directive, governments oblige manufacturers to take the responsibility for the entire life cycle of their products.

Generally, customers have no motivation to return their EOL products after use (Guide *et al*., 2003). Unfortunately, there are few papers that have examined the effect of incentive price on the amount of returned products. The rate of return by the customers is related to the incentive price provided by the collectors, which is determined by various factors, such as distance from the collection or recycling centers and scraps quality. A significant contribution that the present research intends to make and sets it apart from the past researches is that it considers the price effect on the probability amount of returns, taking this fact into account that customers do not have an exact price in mind for selling their products. Therefore, in this paper, the maximum offered price is expressed as a triangular fuzzy number.

This study applies such concepts as distance effect, capacity and penalties simultaneously so that in addition to determining the volume of the products that must be purchased during different periods, it can calculate the optimal number, location, and capacity of the centers to reach the intended goals.

In short, this study attempts to illustrate incentive-dependent returns in dynamic network design, which provides a more realistic expression of the offered incentives effect on customers' reaction, and includes different stages. Furthermore, we will evaluate a new procedure for solving this issue, which benefits from the linearization and an exact solution approach.

The remainder of this paper is organized as follows. After a brief review of the relevant literature in Section 2, in Section 3, we define the problem and introduce the objective functions in detail. Next, the mathematical formulation for the model is developed. In Section 4, a tabu search based heuristic is described for solving the problem. In Section 5, a numerical example of its occurrence is used to show the validation of the heuristic approach and investigate model applicability, following which the computational results are presented for the algorithm. Finally, the concluding remarks are presented in the last section.

2. LITERATURE REVIEW

Logistics literature is remarkably rich in papers in the context of product recovery and recycling. Fleischmann *et al.* (2000) reviewed some case studies on logistics network design for product recovery in different industries, after which he identified general characteristics for such logistics networks. Besides, they denote five groups of activities that appear to be recurrent in EOL product recovery networks: collection, inspection/ separation, reprocessing, redistribution, and disposal.

Additionally, the majority of studies on reverse flows formulate discrete facility location-allocation models. For example, Jayaraman *et al.* (2003) modeled the reverse logistics of hazardous products with a multi-level warehouse location model. The objective of the model is to find the optimal number and location of collection and refurbishing facilities with the corresponding flow of the hazardous products. In their mixed-integer linear programming model, Jayaraman *et al.* (2003) present the number of returned products at each originating site.

Pishvaee *et al.* (2009) classified the literature about the logistics network into three sections: forward logistics, reverse logistics, and integrated forward and reverse logistics. Based on their classified literature, there are not many papers that consider the logistics network design in multi-periods (dynamic) environment. In dynamic environment, Min *et al.* (2006) proposed a dynamic nonlinear mixed-integer programming model for the deterministic logistics network involving both spatial and temporal consolidation of returned products. Furthermore, Ko and Evans (2007) considered a network operated by a third-party logistics service provider. They presented a dynamic mixed-integer nonlinear programming model for the simultaneous design of the forward and reverse network.

In the literature, few papers have attempted to show the relationship between incentives and the product return. Some recent articles are de Figueiredo and Mayerle (2008), Aras and Aksen (2008) and Aksen *et al.* (2009). However, all of them have just considered the location of collection centers and allocation of customer zones to them without appraising cost and revenue issues in multistage nature of reverse logistics network design.

For instance, Aras and Aksen (2008) formulated a mixed-integer nonlinear facility location-allocation model to find both the optimal locations of a predetermined number of collection centers and the optimal incentive values for different return types. In another study, Aksen *et al.* (2009) provided a bi-level mixed-integer nonlinear programming to determine the location of collection centers and the optimal price offered for product return. Clearly, the price offered by the company influences the quality level and the rate of returned products. Therefore, they assume that reservation price follows the right triangular distribution (RTD).

To structure the literature review on logistics network design and to identify future research avenues, we

Table 1. Coding system for classification of logistics network

Specification	Abb.
General structure	
Forward logistics	FL
Reverse logistics	RL
Integrated forward and reverse	IFR
Open loop	OL
Closed loop	CL
Modeling	
Deterministic	
Mixed integer linear programming	MILP
Mixed integer non-linear programming	MINLP
Bi-level mixed integer programming	BLMIP
Stochastic	
Stochastic mixed integer programming	SMIP
Stochastic non-linear programming	SNLP
Robust mixed-integer programming	RMIP
Objectives	
Min cost/Max profit	C/P
Max robustness	Rob
Max responsiveness	Res
Problem definition	
Period	
Single period	SPr
Multi periods	MPr
Product	
Single product	SP
Multi product	MP
Number of facilities to be opened	
Endogenous	En
Exogenous	Ex
Facility	
Capacitated	CF
Uncapacitated	UCF
Demand	
Deterministic	DD
Stochastic	DS
Rate of product return	
Deterministic	RD
Stochastic	RS
Flow	
Capacitated flow	Ca
Uncapacitated flow	Un
Price for returned product	
Price included (should be determined)	PI
Price not included	PN
Network stages	
Supply centers	SC
Production centers	PC
Distribution centers	DC
Collection/inspection centers	CIC
Recycling centers	Ry
Redistribution centers	Rd
Disposal centers	Dp
Remanufacturing	Rm
Repair center	Rp
Refurbishing center	Rf
Disassembly center	Da

have classified design problems of logistics network according to six general specifications: general structure of the network, modeling type, objective functions, problem definition, the network stages included and the solution method. We developed a coding system as shown in Table 1, and the recent mathematical models available in the literature have been coded based on this system in Table 2. Categories based on closed loop or opened loop in Table 2 indicate refer to Fleischmann *et al.* (2000) who expressed the differences between the two networks in their article.

As shown in Table 2, there are few papers that determine the incentive price for product return in reverse logistics network design. These papers have just considered the simplest case of network design, which is comprised of single period, single product, and single stage. Moreover, only the location of collection and inspection centers (CICs) has been determined while other stages of the network have been neglected.

This paper extends the reverse logistics literature by considering the fuzzy relationship between the proposed incentive price and product acquisition. It also considers a dynamic design of reverse logistics with multi-product, multi-quality levels, multi-stage and limited budget in each period. Furthermore, in this research, a tabu search based heuristic has been provided so as to find a solution for the model. The coding of the model under question is provided in Table 2.

3. PROBLEM DEFINITION

The reverse logistics network stages that are considered in this research have been illustrated in Figure 1. Based on this figure, the used products are collected from customer zones and sent to the collection/inspection centers. Afterwards, the products are delivered to different recovery centers for recycling. The following are the assumptions considered in the design of such a network:

- The model is multi-period.
- The model considers multiple products, which have a different but known quality level.
- The potential locations of collection/inspection and recovery facilities are known and fixed.
- Costs parameters (setup, fixed cost, variables, non-utilized capacity, non-collected returns, transportation, and holding costs) are known for each location, product, and time period.
- Holding cost depends on the residual inventory at the end of each period.
- Recovery options operate independently.
- Customers themselves dispose of used products that are not collected by the collectors.
- The budget is only intended for the construction of facilities. It should also be noted that this amount is limited and that the remainder at the end of each period can be used in other periods.

Table 2. Classification of the reviewed literature

Article	General structure	Modeling	Objectives	Problem definition	Network stages	Solution method
Jayaraman *et al.* (2003)	RL, OL	MILP	C	SPr, SP, En, CF, RD, Un, PN	CIC, Rf	Heuristic
Min *et al.* (2006)	RL, CL	MINLP	C	MPr, SP, En, UCF, DD, RD, Un, PN	CIC, Ry	Genetic algorithm
Amiri (2006)	FL, OL	MILP	C	SPr, SP, En, CF, DD, Un, PN	PC, DC	Lagrangian relaxation
Ko and Evans (2007)	IFR, CL	MINLP	C	MPr, MP, En, CF, DD, RD, Un, PN	PC, DC, Rc	Genetic algorithm
Salema *et al.* (2007)	RL, CL, OL	SMIP	C	SPr, MP, En, CF, DS, RS, Un, PN	PC, DC, Da, Dp	Branch and bound
Sharma *et al.* (2007)	RL, CL	MILP	C	MPr, MP, Ex, UCF, DD, Un, PN	Dp, Rf, Da	Exact
Srivastava (2008)	RL, OL	MILP	P	MPr, MP, En, CF, DD, RD, Un, PN	CIC, Rp, Rm, Rf	Exact
Aras and Aksen (2008)	RL, CL	MINLP	P	SPr, MP, Ex, UCF, DD, RS, Un, PI	CIC	Tabu search
Min and Ko (2008)	IFR, OL	MINLP	C	MPr, MP, En, CF, DD, RD, Un, PN	PC, DC, CIC, Rp	Genetic algorithm
de Figueiredo and Mayerle (2008)	RL, OL	BLMIP	P	SPr, SP, En, UCF, RD, Un, PI	CIC, Ry	Heuristic
Cruz-Rivera and Ertel (2009)	RL, CL	MILP	C	SPr, SP, Ex, UCF, DD, Un, PN	CIC	Exact
Aksen *et al.* (2009)	RL,OL	BLMIP	C/P	SPr, SP, Ex, UCF, RS, Ca, PI	CIC	Tabu search
Pishvaee *et al.* (2009)	IFR, CL	SMIP	C	SPr, SP, En, CF, DS, RS, Ca, PN	PC, Ry, DC, CIC, Dp	Exact
El-Sayed *et al.* (2010)	IFR, CL, OL	SMIP	P	MPr, SP, En, CF, DS, RS, Un, PN	SC, PC, DC, Da, Rd, Dp	Exact
Lee *et al.* (2010)	IFR, CL	MILP	C	SPr, MP, En, CF, DS, RS, Un, PN	PC, DC, CIC	Heuristic
Pishvaee *et al.* (2010a)	IFL, CL, OL	MILP	Res, C	SPr, SP, En, CF, DD, RD, Un, PN	PC, DC, CIC, Rm, Dp	Memetic algorithm
Pishvaee *et al.* (2010b)	RL, OL	MILP	Res, C	SPr, SP, En, Ex, CF, RD, Un, PN	CIC, Dp, Rc	Simulated annealing
Pishvaee *et al.* (2011)	IFR, CL	RMIP	C, Rob	SPr, SP, En, CF, DD, RD, Un, PN	Rd, CIC, Dp, Ry	Exact
Abdallah *et al.* (2011)	FL, OL	MIP	C	SPr, MP, En, CF, DD, Un, PN	SC, PC, DC	Exact
Nativi and Lee (2011)	RL, OL	SNLP	C	SPr, SP, Ex, UCF, DS, RS, Un, PN	SC, PC, Ry	Exact
Lamsali (2011)	RL, OL	MINLP	P	SPr, MP, En, CF, RS, Ca, PI	CIC	Exact
Chaabane *et al.* (2012)	IFR, CL	MILP	C	MPr, MP, En, CF, DD, RD, Ca, PN	SC, PC, DC, Ry	Exact
Das and Chowdhury (2012)	RL, CL	MIP	P	SPr, MP, En, CF, DD, RD, Un, PN	PC, DC, SC, Ry, CIC	Exact
Amin and Zhang (2012)	RL, CL	MILP	P	SPr, MP, En, CF, DD, RD, Un, PN	SC, PC, Da, Rf, Dp	Exact
Cardoso *et al.* (2013)	IFR, CL	MILP	P	MPr, MP, En, CF, DS, RS, Un, PN	PC, DC, Dp, CIC, Da	Exact
Eskandarpour *et al.* (2013a)	RL, CL	MILP	Res, C	SPr, MP, En, CF, DD, RD, Ca, PN	PC, CIC, Rp, Dp	Heuristic
Ramezani *et al.* (2013)	IFR, CL	SMIP	Res, P	SPr, MP, En, CF, DS, RS, Ca, PN	SC, PC, DC, CIC, Rm, Dp	Pareto-optimal
Keyvanshokooh *et al.* (2013)	IFR, CL	MILP	C	MPr, MP, En, CF, DD, RS, Un, PI	PC, Ry, DC, CIC, Dp	Exact
Hatefi and Jolai (2013)	IFR, CL	RMIP	C, Rob	SPr, SP, En, CF, DS, RS, Ca, PN	PC, DC, CIC, Rm	Exact
Eskandarpour *et al.* (2013b)	RL, CL	MINLP	Res, C	SPr, MP, En, CF, DD, RD, Ca, PN	SC, CIC, Ry, Dp	Heuristic
Diabat *et al.* (2013)	RL, OL	MINLP	C	SPr, SP, En, CF, DD, RD, Un, PN	CIC	Genetic algorithm Artificial immune system
Our work	RL, OL	FMINLP	C	MPr, MP, En, CF, RS, Ca, PI	CIC, Ry	Tabu search

For abbreviations, see Table 1.
FMINLP: fuzzy mixed-integer nonlinear programming.

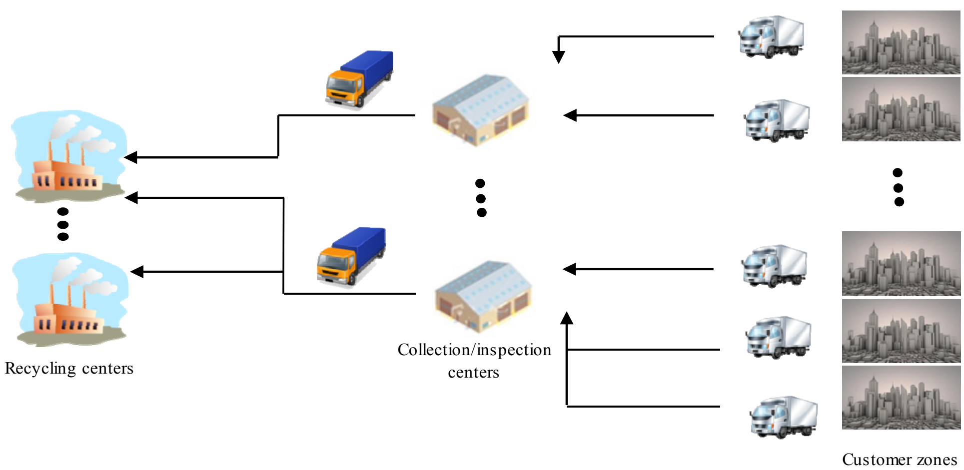

Figure 1. Reverse logistic network.

According to Aksen *et al.* (2009) the product holders' decision to return the products can be modeled by using the notion of consumer surplus. The product holder i which has the used product type p with quality level q at time t will make a return if the collection center offers a unit incentive B_{pqt} that is at least as large as a reservation price B^0_{pqit}. It is assumed that all customers in different locations have the same mental model in responding to the similar offered price. We assume that B^0_{pqit} follows the RTD, whose density function is given in (1) and Figure 2. The proportion P_{pqit} of customers in zone i who are willing to return their used products of type p with quality level of q at time t when the collectors offer incentive B_{pqt} per product is calculated from Eq. (2).

$$f\left(B^0_{pqit}\right) = 2B^0_{pqit} / b^2_{pqit} \tag{1}$$

$$p_{pqit} = \Pr\left(B^0_{pqit} \le B_{pqt}\right) = F\left(B^0_{pqit}\right) = B^0_{pqit} / b^2_{pqit} \tag{2}$$

It is noted that B^0_{pqit} takes values in an interval between 0 and b_{pqit}, which represents the maximum contemplated incentive by a customer for product type p with quality level q. However, considering the deterministic value for the upper bound of this interval may not be suitable since customers do not consider the exact price for the sale of their products. Therefore, we make use of a triangular fuzzy form, defined by $\tilde{b}_{pqit}$, to show the greatest amount possible.

In the next section, based on the aforementioned characteristics of the network, a fuzzy mixed-integer nonlinear programming model is presented.

3.1 Model Formulation

The following notations are used in the formulation of the mentioned design of reverse logistics network:

Sets

P set of used product types
T set of time periods
I set of fixed customer locations
J set of potential CICs
H set of potential recovery locations
L set of transportation mode
M set of recovery facilities (RFs) outputs
N set of capacity levels available for facilities
Q set of used products' quality levels

Parameters

oc_{jt}^n cost of establishing CIC j with capacity level n in period t
fc_{jt} operating fixed cost for CIC j in period t
vc_{pqjt} variable operating cost of product p with quality level q for CIC j in period t
w_{jt}^n capacity with level n for CIC j in period t
hc_{jt} inventory holding cost at CIC j in period t
bc_t available budget for CICs in period t
or_{ht}^n cost of establishing RF h with capacity level n in period t
fr_{ht} operating fixed cost for RF h in period t
vr_{pqht} variable operating cost of product p with quality level q for RF h in period t
pc_{ht} penalty per unit of non-utilized capacity at RF h in period t
car_{ht}^n capacity with level n for RF h in period t
s_{mht} sale price of final product m generated at RF h in period t
α_{mpqht} generation fraction of product m from used product p with quality level q at RF h in period t
br_t available budget for RFs in period t
$d1_i$ distance between customer location i and CIC j
$d2_{jh}$ distance between CIC j and RF h
ct_{lt} handling cost of per unit product traveled in unit distance by transportation mode l in period t
c_{plt} handling cost of product p using transportation mode l in period t
$ca1_{ijlt}$ flow capacity from customer location i to CIC j using transportation mode l in period t
$ca2_{jhlt}$ flow capacity from CIC j to RF h using transportation mode l in period t
r_{pqit} quantity of used product p with quality level q in customer zone i in period t
uc_{pt} penalty per unit of non-collected product p in period t
$\tilde{b}_{pqit}$ contemplated price by customer i for product p with quality level q in period t

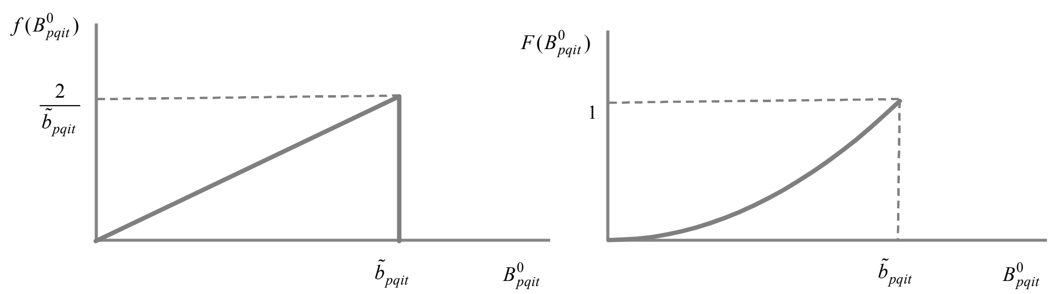

Figure 2. The triangular distribution of purchase incentive.

Decision variables

$Q1_{pqijlt}$ quantity of product p with quality level q shipped from customer zone i to CIC j using transportation mode l in period t

$Q2_{pqjhlt}$ quantity of product p with quality level q shipped from CIC j to RF h using transportation mode l in period t

B_{pqt} offered price for used product p with quality level q in period t

X^n_{jt} binary variable, it takes 1 if CIC j with capacity level n is installed in period t, and 0 if otherwise

Y^n_{ht} binary variable, it takes 1 if RF h with capacity level n is installed in period t, and 0 if otherwise

Dependent variable

I_{pjt} inventory level of used product p collected at CIC j in period t

U_{pit} quantity of uncollected product p from customer i in period t

S_{C_t} remaining budget for CICs at the end of period t

S_{r_t} remaining budget for RFs at the end of period t

p_{pqit} percent of customers in zone i who are willing to return their product of type p with quality level q in period t

φ^n_{jt} binary variable, it takes 1 if CIC j with capacity level n is installed in period t or before t, and 0 if otherwise

ξ^n_{ht} binary variable, it take 1 if RF h with capacity level n is installed in period t or before of t and 0 otherwise

Based on the aforementioned parameters and indices, the fuzzy mixed-integer nonlinear programming (FMINLP) model is developed. The objective function is to minimize the total costs minus potential revenues from recovering the used product. The objective function includes the following issues:

- **Opening and operating cost of collection and inspection centers**: it needs to be mentioned that if the facility opens during a period, it will remain open in the subsequent periods. Establishing the cost of CIC and RF facilities occurs once they are established for the first time. Operating fixed cost is charged in each period after the establishment of the facility.

$$\sum_n\sum_j\sum_t oc^n_{jt}\cdot X^n_{jt}+\sum_j\sum_t fc_{jt}\left(\sum_n \varphi^n_{jt}\right) +\sum_p\sum_q\sum_j\sum_j\sum_l\sum_t vc_{pqit}\cdot Ql_{pqiljt} \tag{3}$$

- **Opening and operating cost of recovery centers:**

$$\sum_n\sum_h\sum_t or^n_{ht}\cdot Y^n_{ht}+\sum_h\sum_t fr_{ht}\left(\sum_n \zeta^n_{ht}\right) +\sum_p\sum_q\sum_j\sum_h\sum_l\sum_t vr_{pqht}Q2_{pqlhlt} \tag{4}$$

- **Purchasing cost of the used products:**

$$\sum_p\sum_q\sum_i\sum_j\sum_l\sum_t Ql_{pqijlt}\cdot B_{pqt} \tag{5}$$

- **Inventory holding cost at CICs:**

$$\sum_p\sum_j\sum_t I_{pjt}\cdot hc_{jt} \tag{6}$$

- **Penalty for non-collected products:**

$$\sum_p\sum_i\sum_t U_{pit}\cdot uc_{pt} \tag{7}$$

- **Transportation cost:**

$$\sum_p\sum_q\sum_i\sum_j\sum_l\sum_t Q1_{pqijlt}\left(c_{plt}+ct_{it}\cdot d1_{ij}\right) +\sum_p\sum_q\sum_j\sum_k\sum_l\sum_t Q2_{pqjhlt}\left(c_{plt}+ct_{it}\cdot d2_{jh}\right) \tag{8}$$

- **Penalty for non-utilization:**

$$\sum_h\sum_t pc_{ht}\cdot\left(\sum_n \zeta^n_{ht}\cdot car^n_{ht}-\sum_p\sum_q\sum_j\sum_l Q2_{pqjhlt}\right) \tag{9}$$

- **Revenues from recovering used product plus the remaining budget in the final period:** these revenues are subtracted from costs at objective function.

$$\sum_m\sum_p\sum_q\sum_j\sum_h\sum_l\sum_t Q2_{pqjhlt}\cdot\alpha_{mpqht}\cdot s_{mht}+Sc_T+S_{r_R} \tag{10}$$

The objective function is to minimize the summation of Eqs. (3)–(9) minus Eq. (10). The constraints for the model are as follows.

$$\sum_t\sum_n X^n_{jt}\le 1 \quad \forall j\in J \tag{11}$$

$$\sum_t\sum_n Y^n_{ht}\le 1 \quad \forall h\in H \tag{12}$$

$$\sum_p\sum_q Q1_{pqijlt}\le\sum_n \varphi^n_{jt}\cdot ca1_{ijlt} \quad \forall i\in I,\forall j\in J,\forall l\in L,\forall t\in T \tag{13}$$

$$\sum_p\sum_q Q2_{pqjhlt}\le\sum_n \xi^n_{ht}\cdot ca2_{jhlt} \quad \forall j\in J,\forall h\in H,\forall l\in L,\forall t\in T \tag{14}$$

$$\sum_p\sum_q\sum_j\sum_h Q2_{pqjhlt}\le\sum_n car^n_{ht}\cdot\xi^n_{ht} \quad \forall h\in H,\forall t\in T \tag{15}$$

$$I_{pj(t-1)}+\sum_q\sum_i\sum_l Q1_{pqijlt}=I_{pjt}+\sum_q\sum_h\sum_l Q2_{pqjhlt} \quad \forall p\in P,\ \forall j\in J,\forall t\in T \tag{16}$$

$$I_{pj0}=0 \quad \forall p\in P,\forall j\in J \tag{17}$$

$$\sum_p I_{pjt} \le w_{jt}^n \cdot \varphi_{jt}^n \quad \forall j \in J, \forall t \in T, \forall n \in N \tag{18}$$

$$\sum_q \sum_j \sum_l Q1_{pqijlt} + U_{pit} = \sum_q r_{pqit} \tag{19}$$
$$\forall p \in P,\ \forall i \in I, \forall t \in T$$

$$\sum_j \sum_l Q1_{pqijlt} \le P_{pqit} \cdot r_{pqit} \tag{20}$$
$$\forall p \in P, \forall q \in Q, \forall i \in I, \forall t \in T$$

$$P_{pqit} = \frac{B_{pqt}^2}{\tilde{b}_{pqit}^2} \quad \forall p \in P, \forall q \in Q, \forall i \in I, \forall t \in T \tag{21}$$

$$\sum_n \sum_j oc_{jt}^n \cdot X_{jt}^n + SC_t = bc_t + Sc_{t-1} \quad \forall t \in T \tag{22}$$

$$Sc_0 = 0, \tag{23}$$

$$\sum_n \sum_h or_{ht}^n \cdot Y_{ht}^n + Sr_t = br_t + Sr_{t-1} \quad \forall t \in T \tag{24}$$

$$Sr_0 = 0, \tag{25}$$

$$B_{pqt} \le \tilde{b}_{pqit} \quad \forall p \in P, \forall q \in Q, \forall i \in I, \forall t \in T \tag{26}$$

$$\sum_{t'=1}^{t} X_{jt'}^n = \varphi_{jt}^n \quad \forall j \in J, \forall n \in N, \forall t \in T \tag{27}$$

$$\sum_{t'=1}^{t} Y_{ht'}^n = \xi_{ht}^n \quad \forall h \in H, \forall n \in N, \forall t \in T \tag{28}$$

$$B_{pqt},\ P_{pqit},\ U_{pit},\ I_{pjt},\ Q1_{pqijlt},\ Q2_{pqjhl},$$
$$Sc_t, Sr_t \ge 0 \quad \forall p \in P, \forall q \in Q, \forall i \in I, \forall j \in H, \tag{29}$$

$$X_{jt}^n, Y_{ht}^n \in \{0, 1\} \quad \forall j \in J, \forall h \in H, \forall n \in N, \forall t \in T \tag{30}$$

Constraints (11) and (12) ensure that a facility can be established in each location at most in one capacity level. Eqs. (13) and (14) represent capacity limitations on product flow between different nodes. Constraint (15) illustrates the capacity restrictions on recovery facilities in terms of the number of total products entering them. Eq. (16) assures the inventory balance of used products at CICs during periods. Eq. (17) determines the initial inventory of products collected at CICs. Constraint (18) indicates the inventory capacity limitation at CICs. In Eq. (19), it is shown that the quantity of returned products is related to the potential quantity in customer zones. On the other hand, this equation bounds the amount of returned products. Constraint (20) represents possible uttermost quantity of various products that can be collected from customer zones. Eq. (21) shows the relationship between the maximum proportion of returns and the offered incentive. Eqs. (22) to (25) assure budget counterbalance among different periods. Fuzzy Eq. (26) guarantees that the offered prices do not exceed the RTD upper bound. Eqs. (27) and (28) specify facilities that have been opened in the previous periods. Finally, constraints in set (29) enforce the non-negativity restrictions on the corresponding decision variables and constraints in set (30) enforce the integrality restrictions on the binary variables.

In this paper, a linear ranking function is applied to convert the fuzzy parameter into the crisp equivalent number through the use of the first index of Yager (1978)

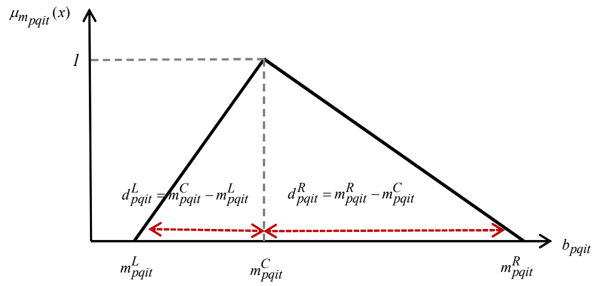

Figure 3. Triangular fuzzy number.

and Yager (1981). Thus, by applying the index of Ronald and considering the triangular fuzzy number of $b_{pqit} = \left(m_{pqit}^L, m_{pqit}^C, m_{pqit}^R\right)$, the aforementioned FMINLP problem is transformed into the crisp equivalent problem by replacing constraints (21) and (26) with the following equations.

$$P_{pqit} = \frac{B_{pqt}^2}{\left[m_{pqit}^C + \dfrac{d_{pqit}^R - d_{pqit}^L}{3}\right]^2}, \tag{31}$$
$$\forall p \in P, \forall q \in Q, i \in I, t \in T,$$

$$B_{pqt} \le m_{pqit}^C + \frac{d_{pqit}^R - d_{pqit}^L}{3} \tag{32}$$
$$\forall p \in P, \forall q \in Q, i \in I, t \in T,$$

where d_{pqit}^R and d_{pqit}^L are the lateral margins (right and left, respectively) of the triangular fuzzy number with central point of m_{pqit}^C (Figure 3).

4. SOLUTION APPROACH

Since our model has been known as NP-hard, it is difficult to obtain high quality solutions for the relatively large-sized examples in a reasonable timeframe using commercial software. Therefore, a heuristic algorithm is developed to solve it.

In this section, we propose an iterative algorithm based on tabu search procedure. Tabu search belongs to a class of local search techniques and is based on avoiding local optima by using memory structures called tabu lists. These lists temporarily record visited solutions and prevent them from being cycled around.

Tabu search performs a search for the solution space by moving from one identified solution to the best solution in a subset of the neighborhood of the current solution. Since there is no necessity to improve the solution in all iterations, a tabu mechanism is provided to prevent the turning process in a sequential series of solutions. This way, the exploration process is not allowed to go back to the previous encountered solutions.

Another feature of the algorithm is called 'diversification' and ensures the search process is not limited to a restricted part of the solution space. In addition, intensification is identified as yet another characteristic of the approach that consists of a greedy local search around the best-recognized solutions.

4.1 Proposed Tabu Search Approach

Due to the type of the model and the dependence of variables, the complexity of this problem is extremely high; this is why it is necessary to use a proper method for providing solutions. In this paper, a heuristic approach is applied, which utilizes two procedures, one being approximate and the other one exact, in each iteration.

We use tabu search for providing solutions to the problem and we take advantage of CPLEX libraries to achieve an exact solution at each stage. First, the model is converted to a convex linear form through determining the neighbor solutions for some variables, including φ, ξ (or X and Y) and B by tabu search algorithm (the full explanation is provided in the initial solution). Afterwards, the model is solved through the use of CPLEX software libraries. Finally, the obtained values by CPLEX for objective functions, *Q1* and *Q2,* return to the main algorithm and this process would be repeated to satisfy the stop condition. We terminate the algorithm as soon as the maximum number of iterations limit is met. The algorithm structure can be seen in Figure 4, where the used notations are defined as follows:

Num_Iter	number of performed iterations
Max_Iter	maximum number of iterations
Num_notchange	number of iterations throughout which the incumbent does not improve
Max_notchange	maximum amount of iterations throughout which the incumbent does not improve
Obj	objective value of newly generated neighboring solution
Best_Neigh	objective value of the best neighboring solution
*Obj**	objective value of the incumbent

The main parts of the algorithm are the following:

Initial solution generation:

As mentioned above, the initial solutions are generated only for the offered price and location of CICs and RFs. These centers are randomly assigned to the available places after reviewing and examining allocation requirements. When the assigning condition, the setup cost is lower than the remaining budget, is satis-

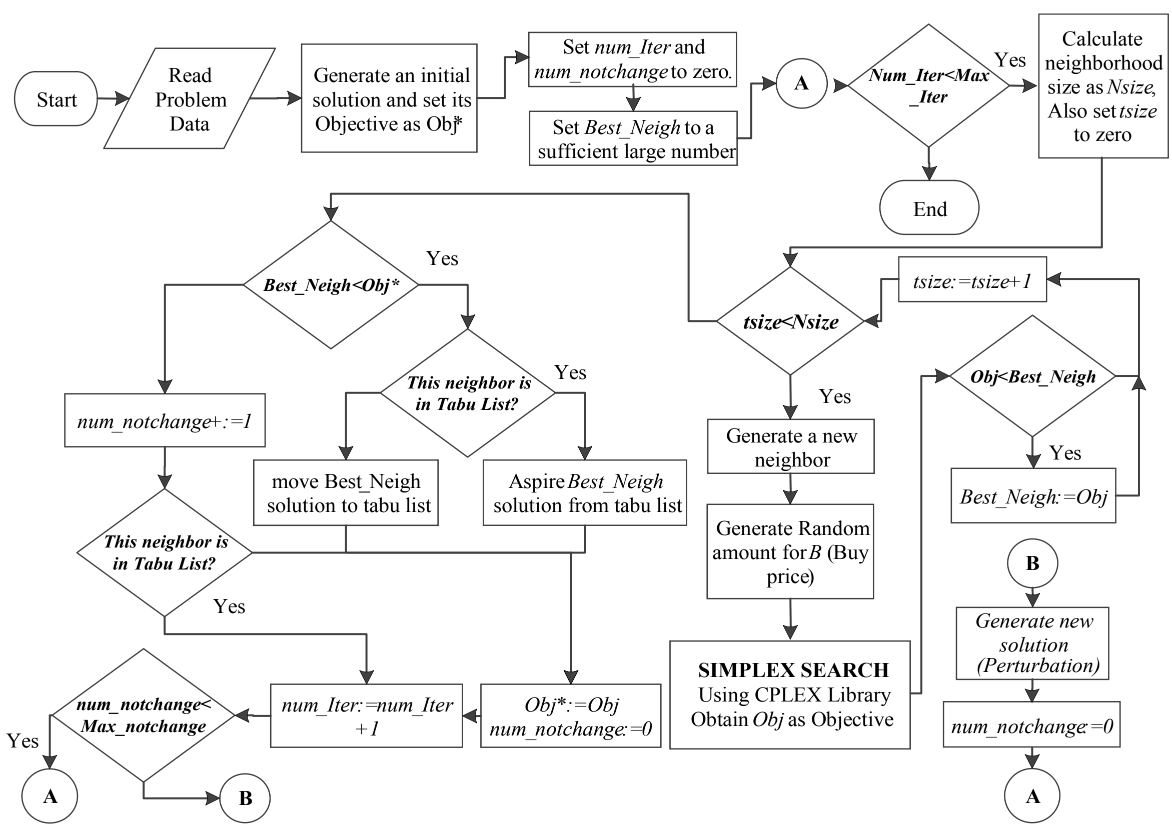

Figure 4. The general structure of proposed tabu search approach.

fied, these places are removed from the list of the available places and the associated variables X and Y, as well as the dependent variables φ and ξ, take 1. Besides, the capacity of centers is randomly selected from among the intended cases for sites (n) in the same way.

Now the amount of B_{pqt} is estimated by a random value between zero and the highest price calculated among customers for the used product p with quality level q in period t.

Generate the neighbor solutions:

At this stage, the entire possible neighborhoods must be investigated. In our algorithm, initially, the location is assigned to the CICs or the RFs without taking the feasibility into account (the procedure is illustrated in Algorithm 1 in Appendix). In Figure 5, a visual representation of generating neighbor solutions is depicted, where rows and columns show potential centers and periods, respectively and black dots indicate construction of the centers.

Afterward their feasibility is considered by functions given in Algorithm 2, and according to what was said about generating initial solutions, the value of offered prices (B) would be determined for every feasible answer. Then, a simplex search will be performed on the model using CPLEX library.

Tabu list:

In any iteration that has achieved a better answer than the existing answers, the current answer is transferred to tabu list to avoid the loop and duplication of answers. This way, the tabu list is specifically designed for objective functions, X and Y. After generating a neighboring solution and determining the objective value, the answer is primarily compared with the current solution. Afterwards, its non-attendance in the tabu list is checked (Algorithm 3). The comparisons are made using the objective values and if they are equal, they will be conducted through the comparison of X and Y.

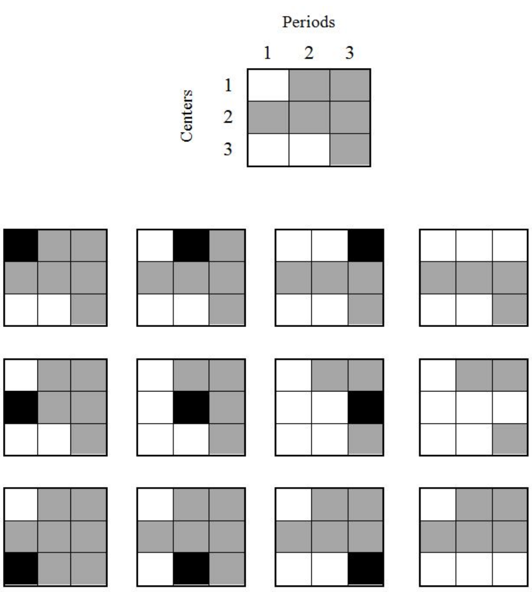

Figure 5. A symbolic representation of generating neighbor solutions.

Releasing condition:

In one of the iterations, if the best solution obtained in the neighborhood is better than the current solution, but has been in the tabu list, it is released from the list and instead, the last solution is listed.

According to the components explained in detail above, the structure of tabu search algorithm is given as Algorithm 4.

Algorithm 4. Main structure of the tabu search algorithm

```
Begin
//TL: Tabu List, contains Collection and Recovery Centers and Fitness;
CapLevCol, CapLevRec: indicate Capacity Level of each established Centers;
Cur, New, N1 and B: indicate Current, New, Best of Neighbors and Best Solution, respectively//

Initialize Solution:
    Cur_φ; Cur_ξ;
    Cur_X; Cur_Y;
    Cur_Q1, Cur_Q2, Cur_Fitness ← Simplex(Cur_ φ, Cur_ ξ);
While iteration < Max_iteration do
    N1Fitness = NB (Cur_ φ, Cur_ ξ);
    If N1_Fitness < Cur_Fitness then
        Tabu_List ← Cur_ φ, Cur_ ξ, Cur_Fitness;
        Cur_Solution ← N1_Solution;
        If TL_Checker (Cur_X, Cur_Y, Cur_Fitness) then
            //Aspiration
            Remove Cur_Solution from TL
        End If.
        If N1_Fitness < Best_Fitness then
            Best_Solution ← N1_Solution;
        End If.
    Else
        If TL_Checker (N1_X, N1_Y, N1_Fitness) then
            Cur_Solution ← N1_Solution;
        End if.
    End If.
    iteration = iteration+1;
End While.
End.
```

5. NUMERICAL EXAMPLE

To evaluate the performance of the proposed approach

Table 3. Test problems' dimensions

		Problem number				
		1	2	3	4	5
Product types	P	2	4	5	3	3
Time periods	T	3	3	4	5	6
Customers	I	5	5	10	20	25
Collection/inspection centers	J	3	4	5	10	15
Recovery centers	H	2	3	3	5	8
Transportation mode	L	2	2	2	2	2
Quality levels	Q	3	3	2	2	2
Capacity levels	N	4	4	2	2	2
Recovery facilities outputs	M	2	2	2	2	2

Table 4. Values of the parameters used in the test problems

Parameter	Symbol	Range*
Setup cost	oc_{jt}^{n}	~ uni[200, 450]
Operating fixed cost	fc_{jt}	~ uni[50, 100]
Operating variable cost	vc_{pqjt}	~ uni[0.1, 1]
Inventory capacity	w_{jt}	~ uni[500, 2000]
Holding cost	hc_{jt}	~ uni[0.2, 0.6]
Budget	bc_{t}	~ uni[400, 700]
Setup cost	or_{ht}^{n}	~ uni[400, 700]
Operating fixed cost	fr_{ht}	~ uni[100, 150]
Operating variable cost	vr_{pqht}	~ uni[0.1, 1.5]
Non-capacity penalty	pc_{ht}	~ uni[0.2, 0.5]
Capacity	car_{ht}^{n}	~ uni[7000, 9000]
Sole price	s_{mht}	~ uni[7, 12]
Transmutation fraction	α_{mpqht}	~ uni[0, 1]
Distance	$d1_{ij}$	~ uni[100, 250]
Distance	$d2_{jh}$	~ uni[40, 150]
Transmit cost	ct_{lt}	~ uni[0.001, 0.01]
Transmit cost	c_{plt}	~ uni[1, 3]
Flow capacity	$ca1_{ijlt}$	~ uni[4000, 6500]
Flow capacity	$ca2_{jhlt}$	~ uni[7000, 9000]
Potential return	r_{pqit}	~ uni[300, 800]
Upper bound price range	$\tilde{b}_{pqit}$	~ uni[5, 9]
Maximum violation	$\tilde{t}_{pqit}$	~ uni[0, 1.6]
Non-collected penalty	uc_{pt}	~ uni[4, 7]

* Uniform distribution[lower bound, upper bound].

on small and medium-sized problems, some illustrative examples have been developed. The size of the investigated examples and their parameters value are specified in Tables 3 and 4, respectively. In Table 4, parameters have been randomly generated using uniform distributions.

Table 5 gives a report of the results obtained by the heuristic method coded in C++ as a sub-algorithm, and numerous reruns at some iteration in the first instance. The resultant behavior of the proposed model is illustrated by a group of titles defined in the following:

1) Is calculated as the sum of the total quantity of uncollected returns from customers,
2) Is the remaining budget for making CICs and RFs at the final date of planned periods,
3) Total income that has been earned by the sale of new products,
4) Total cost is the sum of all costs that are spent in every period of the considered planning horizon,
5) Average time taken by the algorithm for solving the model,

According to the analysis conducted on different iterations, as shown in Table 5, it is observed that there was considerable amelioration among the obtained results although there was not a significant improvement at iterations higher than 200. Figure 6 depicts the objective values that are calculated from the difference between the total costs and incomes.

After this, the approach accuracy assessment results have been summarized using two solving methods, Lingo 13.0 solver and presented tabu search algorithm, in Table 6. All experiments were run in an Intel Core i3 CPU, at 2.13 GHz and with 4.00 GB of RAM. Moreover, the solution of Lingo is a local optimal solution, which can be known from the results in Table 6. To make a comparison, we have used the initial dimensions expressed in Table 3 and have randomly generated ten examples with different values of parameters in the range given in Table 4. According to this table and what

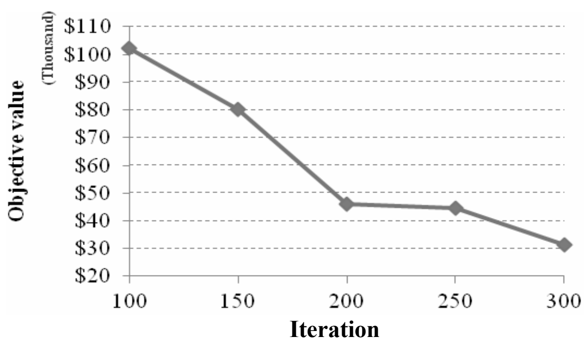

Figure 6. Changes in the initial objective value made by increasing the number of iterations.

Table 5. Changes in the initial output made by increasing the number of iterations

Iteration	Uncollected returns	Remaining budget ($)	Total income ($)	Total cost ($)	Time (sec)
100	28,560	1,068	3,372	105,494.0	321
150	27,659	754	3,058	83,179.2	480
200	18,526	1,986	4,290	50,532.0	656
250	16,583	2,218	4,522	49,127.8	749
300	19,293	1,757	4,061	35,383.7	997

Table 6. Comparison of the results obtained by tabu search and the commercial solver for instance one

Example	Lingo			Tabu search solution		
	Total income ($)	Total cost ($)	Time (sec)	Total income ($)	Total cost ($)	Time (sec)
1	2,050.3	180,896.0	5,884	3,632	123,822.0	602
2	2,165.7	84,959.1	9,416	3,111	73,887.6	659
3	2,409.2	170,351.4	8,971	3,090	120,314.0	643
4	3,440.8	190,282.7	7,463	3,350	113,564.0	638
5	2,247.0	155,448.0	7,543	3,710	68,638.1	643
6	2,061.3	154,646.2	7,090	3,442	88,152.3	632
7	2,059.0	78,565.9	9,724	3,624	69,643.7	666
8	2,564.7	193,548.2	10,133	3,931	69,171.5	644
9	3,202.4	164,761.9	9,241	4,132	91,511.1	617
10	3,574.3	104,940.0	6,074	3,903	75,108.5	639

Table 7. Performance of the solution method on the problem instances

Instance	Uncollected returns	Remaining budget ($)	Total income ($)	Total cost ($)	Time (sec)
(2)	58,241	1,278	9,751	167,914	1,906
(3)	40,687	967	3,768	147,534	806
(4)	157,881	1,374	13,258	704,571	5,529
(5)	146,051	3,190	33,284	560,230	28,358

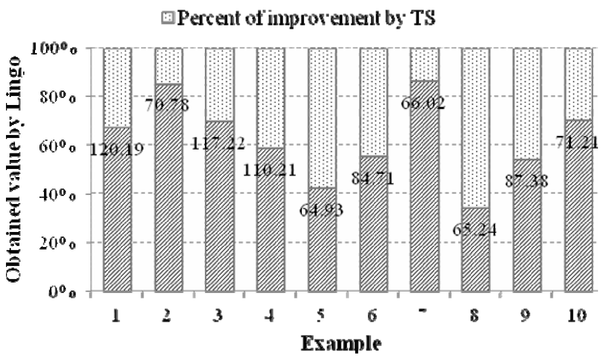

Figure 7. Comparison of objective values by two solving methods for different examples.

is shown in Figure 7, it can be observed that in addition to reducing the time spent on solving the problems through tabu search significantly, the improving results obtained by tabu search for different examples is approximately 10% to 70% better than the Lingo results.

The results of the other instances are presented in Table 7 to show the effectiveness of the proposed tabu search approach in solving higher-dimensional examples. All the runs were done at 200 iterations in the same manner as before. As it can be seen, the suggested approach is able to solve the problems that could not be solved by the commercial software in a finite time, though the solving time has dramatically surged with the increasing size of the problems.

As the results in Table 7 show, it can be understood which heuristic approach causes a significant improvement not only in the average time spent by the algorithm for solving the model but also in the objective values on medium to high-sized problems.

6. CONCLUSION

To sum up, since there are not many articles that discuss financial incentive issues in reverse logistics, this paper investigated a dynamic mathematical model to appraise the incentive effect on return quantity of used products in a reverse logistics network design, which included different stages. In the optimization model, the RTD applying the fuzzy technique was used to interpret this relation. Furthermore, other effective parameters have been applied to configure a mixed integer nonlinear programming model for locating facilities and allocating customers to them, determining the optimal suggested price and planning the recovery strategy in a product return network.

Moreover, a heuristic method based on the tabu search procedure has been proposed for problem solving, which has benefited from the linearization and the exact solution approach. This was followed by comparing the results with those of Lingo commercial software for giving illustrative examples to analyze and validate the method. The computational results show the efficiency and effectiveness of the developed heuristic solution method when time complexity is addressed.

This study has developed several points that can be further investigated in future researches. For instance, the accuracy and efficiency of the proposed method could be improved. Furthermore, a number of verification and validation methods could be helpful in testing the accuracy and consistency of the process. Another potential extension to the setting investigated in this paper may consider the inclusion of different patterns for the behavior responding to the incentives offered. Finally, considering other stages of the reverse logistics network and improving the efficiency of the proposed method may be beneficial in testing the process.

REFERENCES

Abdallah, T., Farhat, A., Diabat, A., and Kennedy, S. (2012), Green supply chains with carbon trading and environmental sourcing: formulation and life cycle assessment, *Applied Mathematical Modelling*, **36** (9), 4271-4285.

Aksen, D., Aras, N., and Karaarslan, A. G. (2009), Design and analysis of government subsidized collection systems for incentive-dependent returns, *International Journal of Production Economics*, **119**(2), 308-327.

Amin, S. H. and Zhang, G. (2012), An integrated model for closed-loop supply chain configuration and supplier selection: multi-objective approach, *Expert Systems with Applications*, **39**(8), 6782-6791.

Amiri, A. (2006), Designing a distribution network in a supply chain system: formulation and efficient solution procedure, *European Journal of Operational Research*, **171**(2), 567-576.

Aras, N. and Aksen, D. (2008), Locating collection centers for distance- and incentive-dependent returns, *International Journal of Production Economics*, **111**(2), 316-333.

Cardoso, S. R., Barbosa-Póvoa, A. P. F., and Relvas, S. (2013), Design and planning of supply chains with integration of reverse logistics activities under demand uncertainty, *European Journal of Operational Research*, **226**(3), 436-451.

Chaabane, A., Ramudhin, A., and Paquet, M. (2012), Design of sustainable supply chains under the emission trading scheme, *International Journal of Production Economics*, **135**(1), 37-49.

Choi, T. M., Li, Y., and Xu, L. (2013), Channel leadership, performance and coordination in closed loop supply chains, *International Journal of Production Economics*, **146**(1), 371-380.

Cruz-Rivera, R. and Ertel, J. (2009), Reverse logistics network design for the collection of end-of-life vehicles in Mexico, *European Journal of Operational Research*, **196**(3), 930-939.

Das, K. and Chowdhury, A. H. (2012), Designing a reverse logistics network for optimal collection, recovery and quality-based product-mix planning, *International Journal of Production Economics*, **135** (1), 209-221.

de Figueiredo, J. N. and Mayerle, S. F. (2008), Designing minimum-cost recycling collection networks with required throughput, *Transportation Research Part E: Logistics and Transportation Review*, **44**(5), 731-752.

Diabat, A., Kannan, D., Kaliyan, M., and Svetinovic, D. (2013), An optimization model for product returns using genetic algorithms and artificial immune system, *Resources, Conservation and Recycling*, **74**, 156-169.

El-Sayed, M., Afia, N., and El-Kharbotly, A. (2010), A stochastic model for forward–reverse logistics network design under risk, *Computers and Industrial Engineering*, **58**(3), 423-431.

Eskandarpour, M., Nikbakhsh, E., and Zegordi, S. H. (2013a), Variable neighborhood search for the bi-objective post-sales network design problem: a fitness landscape analysis approach, *Computers and Operations Research*, **52B**, 300-314.

Eskandarpour, M., Zegordi, S. H., and Nikbakhsh, E. (2013b), A parallel variable neighborhood search for the multi-objective sustainable post-sales network design problem, *International Journal of Production Economics*, **145**(1), 117-131.

Fleischmann, M., Krikke, H. R., Dekker, R., and Flapper, S. D. P. (2000), A characterisation of logistics networks for product recovery, *Omega*, **28**(6), 653-666.

Govindan, K., Palaniappan, M., Zhu, Q., and Kannan, D. (2012), Analysis of third party reverse logistics provider using interpretive structural modeling, *International Journal of Production Economics*, **140** (1), 204-211.

Guide Jr, V. D. R., Teunter, R. H., and Van Wassenhove, L. N. (2003), Matching demand and supply to maximize profits from remanufacturing, *Manufacturing and Service Operations Management*, **5**(4), 303-316.

Hatefi, S. M. and Jolai, F. (2014), Robust and reliable forward–reverse logistics network design under demand uncertainty and facility disruptions, *Applied Mathematical Modelling*, **38**(9), 2630-2647.

Ilgin, M. A. and Gupta, S. M. (2010), Environmentally conscious manufacturing and product recovery (EC MPRO): a review of the state of the art, *Journal of Environmental Management*, **91**(3), 563-591.

Jayaraman, V., Patterson, R. A., and Rolland, E. (2003), The design of reverse distribution networks: models and solution procedures, *European Journal of Operational Research*, **150**(1), 128-149.

Keyvanshokooh, E., Fattahi, M., Seyed-Hosseini, S. M., and Tavakkoli-Moghaddam, R. (2013), A dynamic pricing approach for returned products in integrated forward/reverse logistics network design, *Applied Mathematical Modelling*, **37**(24), 10182-10202.

Ko, H. J. and Evans, G. W. (2007), A genetic algorithm-based heuristic for the dynamic integrated forward/ reverse logistics network for 3PLs, *Computers and Operations Research*, **34**(2), 346-366.

Krikke, H., le Blanc, I., van Krieken, M., and Fleuren, H. (2008), Low-frequency collection of materials disassembled from end-of-life vehicles: on the value of on-line monitoring in optimizing route planning, *International Journal of Production Economics*, **111** (2), 209-228.

Lamsali, H. (2012), A mixed integer non-linear programming model for optimizing the collection methods of returned products, *Proceedings of the 2011 9th International Conference on ICT and Knowledge Engineering*, Bangkok, Thailand, 129-133.

Lee, C., Realff, M., and Ammons, J. (2011), Integration of channel decisions in a decentralized reverse production system with retailer collection under deterministic non-stationary demands, *Advanced Engineering Informatics*, **25**(1), 88-102.

Lee, D. H., Dong, M., and Bian, W. (2010), The design of sustainable logistics network under uncertainty, *International Journal of Production Economics*, **128**(1), 159-166.

Min, H. and Ko, H. J. (2008), The dynamic design of a reverse logistics network from the perspective of third-party logistics service providers, *International Journal of Production Economics*, **113**(1), 176-192.

Min, H., Ko, C. S., and Ko, H. J. (2006), The spatial and temporal consolidation of returned products in a closed-loop supply chain network, *Computers and Industrial Engineering*, **51**(2), 309-320.

Nativi, J. J. and Lee, S. (2012), Impact of RFID information-sharing strategies on a decentralized supply chain with reverse logistics operations, *International Journal of Production Economics*, **136**(2), 366-377.

Östlin, J., Sundin, E., and Björkman, M. (2008), Importance of closed-loop supply chain relationships for product remanufacturing, *International Journal of Production Economics*, **115**(2), 336-348.

Pishvaee, M. S., Farahani, R. Z., and Dullaert, W. (2010a), A memetic algorithm for bi-objective integrated forward/reverse logistics network design, *Computers and Operations Research*, **37**(6), 1100-1112.

Pishvaee, M. S., Jolai, F., and Razmi, J. (2009), A stochastic optimization model for integrated forward/ reverse logistics network design, *Journal of Manufacturing Systems*, **28**(4), 107-114.

Pishvaee, M. S., Kianfar, K., and Karimi, B. (2010b), Reverse logistics network design using simulated annealing, *International Journal of Advanced Manufacturing Technology*, **47**(1-4), 269-281.

Pishvaee, M. S., Rabbani, M., and Torabi, S. A. (2011), A robust optimization approach to closed-loop supply chain network design under uncertainty, *Applied Mathematical Modelling*, **35**(2), 637-649.

Pokharel, S. and Liang, Y. (2012), A model to evaluate acquisition price and quantity of used products for remanufacturing, *International Journal of Production Economics*, **138**(1), 170-176.

Ramezani, M., Bashiri, M., and Tavakkoli-Moghaddam, R. (2013), A new multi-objective stochastic model for a forward/reverse logistic network design with responsiveness and quality level, *Applied Mathematical Modelling*, **37**(1), 328-344.

Salema, M. I. G., Barbosa-Povoa, A. P., and Novais, A. Q. (2007), An optimization model for the design of a capacitated multi-product reverse logistics network with uncertainty, *European Journal of Operational Research*, **179**(3), 1063-1077.

Schultmann, F., Zumkeller, M., and Rentz, O. (2006), Modeling reverse logistic tasks within closed-loop supply chains: an example from the automotive industry, *European Journal of Operational Research*, **171**(3), 1033-1050.

Sharma, M., Ammons, J. C., and Hartman, J. C. (2007), Asset management with reverse product flows and environmental considerations, *Computers and Operations Research*, **34**(2), 464-486.

Srivastava, S. K. (2008), Network design for reverse logistics, *Omega*, **36**(4), 535-548.

Tagaras, G. and Zikopoulos, C. (2008), Optimal location and value of timely sorting of used items in a remanufacturing supply chain with multiple collection sites, *International Journal of Production Economics*, **115**(2), 424-432.

Wojanowski, R., Verter, V., and Boyaci, T. (2007), Retail–collection network design under deposit–refund, *Computers and Operations Research*, **34**(2), 324-345.

Yager, R. R. (1978), Ranking fuzzy subsets over the unit interval, *Proceedings of IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, San Diego, CA, 1435-1437.

Yager, R. R. (1981), A procedure for ordering fuzzy subsets of the unit interval, *Information Sciences*, **24**(2), 143-161.

Appendix

Algorithm 1. Neighborhood solution generating procedure

```
Procedure NB (φ, ξ):
Begin

For each j in J and t in T
   New_ φ ← φ;
   New_X ← Cur_X;
   Capacity_Level ← Random number between zero and
   Capacity Level Number;
   For each t' in T
      If t' < t then
         New_ φ [Capacity_Level][j][t'] = False;
      Else
         New_ φ [Capacity_Level][j][t'] = True;
      End If.
      New_X [Capacity_Level][ j][t'] = False;
   End Loop.
   New_X [Capacity_Level][j][t'] = True;
   If Feasibility_Checker_Col (New_X)
      New_Q1, New _Q2, New_Fitness ← Simplex
      (New_ φ, ξ);
      If New_Fitness < N1_Fitness
         N1_Solution ← New_Solution;
      End If.
   End If.
End Loop.

For each h in H and t in T
   New_ ξ ← ξ;
   New_Y ← Cur_Y;
   Capacity_Level ← Random number between 0 and
   Capacity Level Number;
   For each t' in T
      If t' < t then
         New_ ξ [Capacity_Level][h][t'] = False;
      Else
         New_ ξ [Capacity_Level][h][t'] = True;
      End If.
      New_Y [Capacity_Level][j][t'] = False;
   End Loop.
   New_Y [Capacity_Level][j][t'] = True;
   If Feasibility_Checker_Rec (New_Y)
      New_Q1, New _Q2, New_Fitness ← Simplex (φ,
      New_ ξ);
      If New_Fitness < N1_Fitness
         N1_Solution ← New_Solution;
      End If.
   End If.
End Loop.

Return N1_Fitness;
End.
```

Algorithm 2. Feasibility checker procedure

```
Procedure Feasibility_Checker_Col (New_X):
Begin
feasibility: Binary variable;
For each t in T
   NewSC [t] = BC[t] + NewSC [t-1];
   If NewSC [t-1] <0 then
      feasibility = false;
      break;
   End If.
   For each j in J
      If New_X [j][t]
         NewSC [t] -= OC [CapLevCol][j]][j][t];
      End If.
   End Loop.
End Loop.

Return feasibility;
End.
```

```
Procedure Feasibility_Checker_Rec (New_Y):
Begin
feasibility: Binary variable;
For each t in T
   NewSR [t] = BR[t] + NewSR [t-1];
   If NewSR [t-1] <0 then
      feasibility = false;
      break;
   End If.
   For each h in H
      If New_Y [h][t]
         NewSR [t] -= OR [CapLevRec][h]][h][t];
      End If.
    End Loop.
End Loop.

Return feasibility;
End.
```

Algorithm 3. Tabu List checker Procedure

```
Procedure TL_Checker(X, Y, Fitness):
Begin
Status: Binary variable;
Status = False;
For each solution i in TL
    If TL.Fitness[i] == Fitness Then
        For each t in T
            For each j in J
                If TL.Col[i][j][t] != X[j][t] Then
                    Status = True;
                    break;
                End If.
            End Loop.
            For each h in H
              If TL.Rec[i][h][t] != Y[h][t] Then
                  Status = True;
                  break;
              End If.
          End Loop.
      End Loop.
  End If.
End Loop.

Return Status;
End.
```