Facility Location Optimization Using a Hybrid Model of Disassembly Line Balancing and Closed-Loop Supply Chain

Mostafa Danesh, Semperger Sandor and Amir Mosavi

January 18, 2020

Facility Location Optimization Using a Hybrid Model of Disassembly Line Balancing and Closed-Loop Supply Chain

Mostafa Danesh [1], Semperger Sandor [2], Amir Mosavi [3,4*]

[1] Department of Industrial Engineering, University of Kurdistan, Pasdaran St.1, Sanandaj-66177-15175, Iran; m.danesh@eng.uok.ac.ir

[2] Institute of Automation, Kando Kalman Faculty of Electrical Engineering, Obuda University, 1034 Budapest, Hungary; semperger.sandor@kvk.uni-obuda.hu, amir.mosavi@kvk.uni-obuda.hu

[3] Institute of Research and Development, Duy Tan University, Da Nang 550000, Vietnam; amirhoseinmosavi@duytan.edu.vn

[4] Institute of Structural Mechanics, Bauhaus Universität-Weimar, Marienstraße 15, D-99423 Weimar, Germany; amir.mosavi@uni-weimar.de

Abstract: This study investigates a hybrid facility location computation of closed-loop supply chain network to optimize the cost. A mixed-integer linear programming model is proposed and implemented using general algebraic modeling system (GAMS). Considering the NP-Hardness and the exponential growth of solution time by increasing its size, a genetic metaheuristic algorithm is proposed for generalization. The performance evaluation of the model shows promising results.

Keywords: Closed-loop supply chain (CLSC); disassembly line balancing (DLB); facility location; genetic algorithm (GA); linear coding; reverse logistics

1 Introduction

The supply chain is referred to a network of moving of materials or products within the hands of customers, retailers, distributors, manufacturers, and suppliers. Considering the growing environmental concerns, effective utilization of raw materials has become challenging using only the traditional supply chains. As an alternative, in a closed-loop supply chain (CLSC), the major part of raw materials are to be supplied through recycling and reverse supply chains. In the forward supply chain, raw materials are brought from suppliers and then manipulated in production or assembly processes and finally given to customers via distribution centers. In the reverse supply chain, a part of productions is recycled by collection centers and sent to repair and disassembly centers and finally considering the carried out processes, worthless parts are sent to dumping centers and final, and half-built parts are directed to forward supply chains (Schultmann et al. 2006). Based on this concept, closed-loop supply chain management is the design, verification, and usage of a system. Its aim is maximizing the life cycle value of a product with a dynamic return of the value in several kinds and volumes over time. Figure 1 describes the CLSC in general. Fill and hachure lines introduce forward and reverse supply chain, respectively.

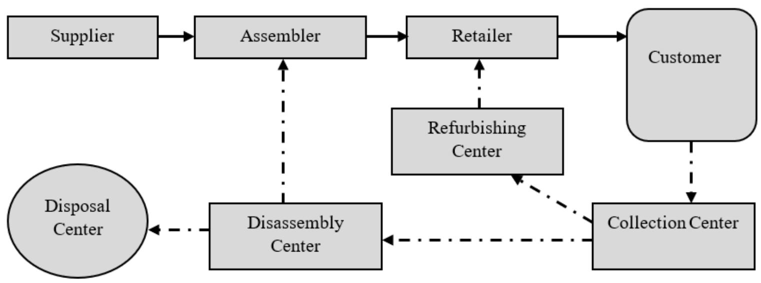


Figure 1. Overview of closed-loop supply chain

Disassembly process is the first phase in the product recycling operation. Disassembly is the main activity in reverse supply chain which can reuse and recycle products and prevent their waste. It is a systematic method to divide the product into their components, subassemblies or other categories (Altekin et al.2008). Disassembly line balancing (DLB) is defined as assigning disassembly activities while satisfying priority of needs to the work stations so that all transposition relationships are satisfied. Minimum number of work stations and cycle time can be obtained by enforcing disassembly line balance constraints. This topic has recently been considered in CLSC networks in detail (Gungor and Gupta, 2008; Ilgın and Gupta, 2010; Ding et al.2010). To achieve a high-speed and competitive closed-loop supply chain, reverse distribution processes and disassembly lines should be able to work simultaneously (Fleischmann et al., 2001, 2004). By integrating decisions in strategic, tactical and operational levels in closed-loop supply chain design, models could be optimized in terms of maximizing customer satisfaction, minimizing the total costs and etc. The integration of disassembly and distribution processes results in greater accuracy, control, and reliability (Ozceylan, 2014). Deciding on the location of the facilities has an impact in designing the CLSC networks (Ilgın and Gupta, 2010; Melo et al., 2009; Hassanzadeh Amin and Baki, 2013; Ghadge et al. 2016). The usefulness and applicability of locating, especially in logistics is undeniable and this is one of the most effective logistical decisions in the supply chain. Nowadays low attention has been given to the relationship between facility location problem at the strategic level and the integrated subject of closed-loop supply chain network and disassembly line balancing in tactical and operational levels which is the subject of the current research. This research would bring the problem closer to the real-world situation. Presenting this model can simultaneously optimize factors including cycle time, total costs, number of workstations located on the disassembly line, number of subassemblies purchased by suppliers and the number of the subassemblies or the final products dispatched between the facilities.

In the following, the paper is arranged as below. In Section 2, the integration issues publications in the CLSE will be browsed according to different decision levels. In section 3, a nonlinear mathematical model is introduced for a CLSC network design. A numerical sample is proposed in section 4. The proposed model will be examined using GAMS software in section 5 and also a metaheuristic method is developed for large scale problems in this section. Finally, the conclusion and future trends of the research are described in section 6.

2 Literiture review

Agrawal and Tiwari (2008) presents the elements of an optmial closed-loop supply chain where the products in forward and backward chains and tactical level are related to disassemble line balance in reverse chain. In this research which utilizes ants' colony optimization solution, minimizing cost of transportation, purchases, repair, and disassembly work stations' operation is automatically considered by focusing on minimizing the number of work stations and idle time during a given cycle time. Altkin and Akkan (2012) presented a predictive-reactive approach for disassemble line balance problem failure activities in order to enhance profitability. Creating a significant uncertainty in disassembly activities is known as activities' failure probability. These failures can make next-level activities impossible and also change the downstream stations' capacity. Firstly, a predictive balance is made to guarantee a specific performance level, in the second step failure activities are again chosen and assigned to the stations using a balancing model in order to recover what was lost just like a line that should be balanced again. This study shows that using the integrated approach, a set of activities should be simultaneously specified while lines' balancing is done for a number of alternative stations.

In the research conducted by Amin and Zhang (2012), a general closed-loop supply chain network including producer, disassemble, repair, and dump locations is studied which is managed by the producer. In this research, an integrated two-phase model is presented that an infrastructure is presented for supplier selection criterion in reverse logistics in the first step. Furthermore, according to qualitative criterion for supplier's evaluation, a fuzzy method is modeled. The product of this phase is each supplier's weighting based on each section. Manzini and Bindi (2009) presented a novel efficient way to integrate the decisions made in classification and management of logistics network which includes the decision about number and location of facilities, assigning public demand points to existing suppliers, selecting the transportation method and travels' optimization. In other words, in this research routing of daily trips is considered as operational level, assigning customers to distributers is known as tactical level and location of facilities is the strategic level. The purpose of this article is to form a main infrastructure for designing and optimizing a multi-category and multi-level production/distribution system. The solution used in it is a mixture of integer linear programming modeling and heuristic algorithms, cluster analysis and also optimal transportation rules. According authors of this article, implementing this case study in America's tile industry will have at least 11 percent saving in logistics costs, which are some thousand dollars per year. In the research conducted by Miranda and Garrido (2004), a concurrent method is used to combine inventory control choices (such as economic order quantity and assurance inventory storage) and public facility location models. Its aim is to solve distribution network design problems. The used method in this analysis is a combination of mixed-integer non-linear programming model and a heuristic solution according to lagrangian relaxation and sub-gradient method to find the proper reductive direction, which is known as DNDRP. In this research, we consider the inventory control as a level of functional-tactical and also distribution network design as a level of strategic. In the numerical application, it is demonstrated that potential cost reduction will have a higher raise than the traditional approaches when inventory costs and/or demand changes are higher. In a research by McGovern and Gupta (2007) a disassembly line balance problem is introduced and devised through combinational optimization ways, and it is considered as a complete search that presents the optimal solution in a regular form. Due to the exponential time complexity of the problem, the speed is reduced by increasing the problem size.

In a research by Ozceylan and Paksoy (2013) balancing of opened work stations was conducted concurrently for optimization of reverse supply chain including costumers, collection/disassembly centers and factories with the goal of minimizing transportation costs and to stabilize the total costs for disassembly lines. The important purpose of this article is to introduce and specify the reverse supply chain design integration problem and the disassembly line balance problem. The solution used in current work is a complex integer non-linear programming method that works concurrently on determining optimal distribution among less cost facilities, determining disassembly line works stations numbers with least opening cost, specifying the time of cycle of any disassembly center and optimal assigning of activities to work stations too. In fact, the proposed mathematical model, which includes collection, dump, and disassembly processes searches the optimal distribution network in a manner that the disassemble lines in each disassembler are balanced simultaneously. We consider the disassembly line balance issue as a tactical level and reverse supply chain design as a strategic level. This research shows that the demand for disassembled products in reverse supply chain networks design is a key decision variable. In the research by Salema et al. (2009), a strategic location-assignment method is extended for forward or reverse supply chains, concurrent model.

Kaya and Urek (2015) and Paydar et al. (2017) addressd the inventory and pricing issues in CLSC, while Hassanzadeh et al. (2016) considered exchange rates and customs. Al-Salem et al. (2016) considered a mixed-integer and non-linear for warehouses location. Xu et al. (2016) analyzed the effect of carbon emissions consideration on the design of both hybrid and dedicated CLSCs. Jabbarzadeh et al. (2017) and Bhattacharyya et al. (2017) examined a stochastic model of CLSC for supply chain cost. Ahmadzadeh and Vahdani (2017) investigate three meta-heuristic algorithms poterntial with promissing results. Soleimani et al. (2017), designed a novel CLSC for considering environmental responsibilities. Gen-Han Wu et al. (2018) and Yavari and Geraeli (2019) proposed a mixed-integer linear programming model using two-stage interactive possibilistic. Cheraghalipour et al. (2018) proposed a novel bi-objective five-echelon CLSC for citrus to minimize the total network. Jerbia et al. (2018) and Vahdani and Ahmadzadeh (2019) established a hybrid integer nonlinear programming model for ICT products chain. Hajipour et al. (2019) and Zhen et al. (2019) presented a multi-echelon, multiproduct and emissions control model for the CLSC. As a response to the above literiture, in this paper, integration of tactical and strategic decisions is implemented in terms of two small and large continuous time scales. In the large scale the supply chain is designed to calculate demand and existing returns while satisfaction is planned simultaneously and also tactical decisions are made in the small scale. A complex integer linear programming formulation is resulted that will be a solution for optimization using branch and bound techniques. In this research, we consider production/warehouse programming and plan of closed-loop supply chain network in the forms of tactical level and strategic level respectively.

3 Problem definition and modeling

Recently optimization of a CLSC, as a part of the green supply chain, has been more considered by researchers (Krikke et al., 2013; Stindt and Sahamie, 2014; Fleischmann et al., 1997; Daniel et al., 2009; Akcali and Cetinkaya, 2011; Govindan et al., 2013; Govindan et al., 2013). In this section, we offer a new single objective way as a mixed integer programming aimed at keeping down the entire cost of CLSC. The model considers locating two sets of refurbishing and disassembly facilities at the strategic level, and DLB at the tactical level with execution of the necessary constraints. So the new model considers location of facilities in the integrated problem of CLSC network and DLB. It is assumed that a fixed number of potential centers are available for refurbishing and disassembly facilities. In this section first the suggested mathematical model is presented. Then linearization operations are applied on some of constraints considered in this model.

3.1 Notation and mathematical formulation

In current model a closed-loop supply chain is made up of two parts, as forward and reverse supply chains. First, the purchase of subassemblies from suppliers, assembling them to final products and delivering to final consumer is done in forward supply chain. Then the collection and repair of products from final consumers and also disassembly and dumping worthless subassemblies are done in reverse supply chains. There are two options for supplying subassemblies. The first option is the main suppliers and the second one is the disassemblers. There are also two options for supplying the final product. The first one is the assemblers and the second one is the repair centers. In fact the presence of suppliers that allows the supply of products or subassemblies at a lower cost to the reverse supply chain directly makes the CLSC occurs as forward and reverse chains dependent.)

So integrating CLSC and DLB is a technique to relate forward and reverse supply chains. here, the quantities dispatched between subassemblies and products between the levels of the supply chain, cycle time, opened workstations, reconstruction and disassemblies centers, allocating disassembly activities to the workstations and allocation or non-allocation of facilities to the customers are concepts of output variables in this model. The assumptions of the problem modeling are such as Ozceylan et al. (2014) and also it is supposed that several candidate sites which are considered for establishment of repair and disassembly facilities and location of other facilities are assumed fixed and known. Index sets, suppliers: I={1,2,…}, assemblers: J={1,2,…}, retailers: K={1,2,…}, customers: L={1,2,…}, collection centers: M={1,2,…}, furbishing centers: R={1,2,…}, disassemblers: D={1, 2,…}, period: P={1, 2,…}, subassemblies: C={1, 2,…}, disassembly stations: S={1, 2, …}, disassembly tasks: T={1, 2, …}, artificial nodes: A= {0, 1, ...}, where, the integer and binary variables and also Parameter in ref.1. Some f them are as below: Y_rp: If the refurbished center r∈R opens in p∈P period, it will be one; otherwise it will be zero. Y_dp : If the disassembly center d∈D opens in p∈P period, it will be one; otherwise it will be zero. X_rkp : If the reconstruction center r∈R gives service to retailer k∈Kth during p∈P period, it will be one; otherwise it will be zero. X_djp: If disassembly center d∈D gives service to assembler j∈J during p∈P period, it will be one; otherwise it will be zero, r_3 stands for number of potential sites of refurbishing centers, and r_4 represents number of potential sites of disassembly centers. Considering the notation introduced, the considered issue can be expressed as a non-linear programming structure model of mixed integer number as Eqs. (1-29)

$$\begin{aligned} Min\ \boldsymbol{t}(&\textstyle\sum_{i\in I}\sum_{j\in J}\sum_{c\in C}\sum_{p\in P}X_{ijcp}\,d_{ij}+\sum_{j\in J}\sum_{k\in K}\sum_{p\in P}Y_{jkp}\,d_{jk}+\\ &\textstyle\sum_{k\in K}\sum_{m\in M}\sum_{p\in P}W_{klp}\,d_{kl}+\sum_{l\in L}\sum_{m\in M}\sum_{p\in P}A_{lmp}\,d_{lm}+\\ &\textstyle\sum_{m\in M}\sum_{r\in R}\sum_{p\in P}B_{mrp}\,d_{mr}+\sum_{m\in M}\sum_{d\in D}\sum_{p\in P}S_{mdp}\,d_{md}+\\ &\textstyle\sum_{r\in R}\sum_{k\in K}\sum_{p\in P}E_{rkp}\,d_{rk}+\sum_{d\in D}\sum_{j\in J}\sum_{c\in C}\sum_{p\in P}Z_{djcp}\,d_{dj}+\\ &\textstyle\sum_{d\in D}\sum_{c\in C}\sum_{p\in P}F_{dcp}\,d_{dc})+\sum_{i\in I}\sum_{j\in J}\sum_{c\in C}\sum_{p\in P}X_{ijcp}\,S_{ic}+\\ &\textstyle\sum_{m\in M}\sum_{r\in R}\sum_{p\in P}B_{mrp}\,W_r+\sum_{s\in S}\sum_{d\in D}\sum_{p\in P}N_{sdp}\,O_{dp}+\sum_{r\in R}\sum_{p\in P}F_{rp}\,Y_{rp}+\\ &\textstyle\sum_{d\in D}\sum_{p\in P}F_{dp}\,Y_{dp} \end{aligned} \tag{1}$$

$\sum_{i\in I} X_{ijcp} \leq a_{icp} \qquad \forall\, i \in I, c \in C, p \in P$ (2)

$\sum_{j\in J} Y_{jkp} \leq b_{jp} \qquad \forall\, j \in J, p \in P$ (3)

$\sum_{l\in L} W_{klp} \leq c_{kp} \qquad \forall\, k \in K, p \in P$ (4)

$\sum_{k\in K} W_{klp} \geq u_{lp} \qquad \forall\, l \in L, p \in P$ (5)

$\sum_{r\in R} B_{mrp}.Y_{rp} + \sum_{d\in D} S_{mdp}\,.Y_{dp} \leq e_{mp} \qquad \forall\, m \in M, p \in P$ (6)

$\sum_{k\in K} E_{rkp}\,.X_{rkp} \leq f_{rp} \qquad \forall\, r \in R, p \in P$ (7)

$F_{dcp}.X_{djp} + \sum_{l\in L} Z_{djcp}\,.X_{djp} \leq g_{dcp} \qquad \forall\, d \in D, c \in C, p \in P$ (8)

$\sum_{i\in I} X_{ijcp} + \sum_{d\in D} Z_{djc(p-1)}\,.X_{dj(p-1)} = r_c \sum_{k\in K} Y_{jkp} \qquad \forall\, j \in J, c \in C, p \in P$ (9)

$\sum_{j\in J} Y_{jkp} + \sum_{r\in R} E_{rk(p-1)}\,.X_{rk(p-1)} = \sum_{l\in L} W_{klp} \qquad \forall\, k \in K, p \in P$ (10)

$\theta_{min} \sum_{k\in K} W_{klp} \leq \sum_{m\in M} A_{lmp} \leq \theta_{max} \sum_{k\in K} W_{klp} \qquad \forall l \in L, p \in P$ (11)

$\lambda \sum_{l\in L} A_{lmp} = \sum_{r\in R} B_{mrp}.Y_{rp} \qquad \forall m \in M, p \in P$ (12)

$\sum_{m\in M} B_{mrp} = \sum_{k\in K} E_{rkp} \qquad \forall\, r \in R, p \in P$ (13)

$(1-\lambda) \sum_{\mathrm{l}\in \mathrm{L}} \mathrm{A}_{\mathrm{lmp}} = \sum_{\mathrm{d}\in \mathrm{D}} \mathrm{S}_{\mathrm{mdp}}.\mathrm{Y}_{\mathrm{dp}} \qquad \forall \mathrm{m} \in \mathrm{M}, \mathrm{p} \in \mathrm{P}$ (14)

$r_c(1-\mu) \sum_{m\in M} S_{mdp}.X_{djp} = F_{dcp} \qquad \forall\, d \in D, c \in C, p \in P$ (15)

$r_c.\mu \sum_{m\in M} S_{mdp}.X_{djp} = \sum_{j\in J} Z_{djcp} \qquad \forall\, d \in D, c \in C, p \in P$ (16)

$\sum_{t:B_t\in S(A_a)} L_{tdp} = 1 \qquad \forall\, a = 0, d \in D, p \in P$ (17)

$\sum_{t:B_t\in S(A_a)} L_{tdp} = \sum_{t:B_t\in P(A_a)} L_{tdp} \qquad \forall\, a \neq 0 \in A, d \in D, p \in P$ (18)

$\sum_{s\in S} M_{tsdp} = L_{tdp} \qquad \forall\, t \in T, d \in D, p \in P$ (19)

$\sum_{t:B_t\in P(A_a)} \sum_{s\in S} M_{tsdp} \geq \sum_{t:B_t\in S(A_a)} M_{tvdp} \qquad \forall\, a \neq 0 \in A, v \in J, d \in D, p \in P$ (20)

$\sum_{t\in T} M_{tsdp}.d_{B_t} \leq (W_{time}/(\sum_{j\in J}\sum_{c\in C} Z_{djcp} + \sum_{c\in C} F_{dcp})N_{sdp}) \qquad \forall\, s \in S, d \in D, p \in P$ (21)

$\sum_{r\in R} X_{rkp} \leq r_3 \qquad \forall\, r \in R, k \in K, p \in P$ (22)

$\sum_{d\in D} X_{djp} \leq r_4 \qquad \forall\, d \in d, j \in J, p \in P$ (23)

$X_{djp} \leq Y_{dp} \qquad \forall\, d \in D, j \in J, p \in P$ (24)

$X_{rkp} \leq Y_{rp} \qquad \forall\, r \in R, k \in K, p \in P$ (25)

$\sum_{k\in K} X_{rkp} \geq Y_{rp} \qquad \forall\, k \in K, r \in R, p \in P$ (26)

$\sum_{j\in J} X_{djp} \geq Y_{dp} \qquad \forall\, j \in J, d \in D, p \in P$ (27)

$X_{ijcp}, Y_{jkp}, W_{klp}, A_{lmp}, B_{mrp}, S_{mdp}, E_{rkp}, Z_{djcp}, F_{dcp}, CT_{dp} \geq 0$ (28)
$\forall\, i \in I, j \in J, k \in K, l \in L, m \in M, r \in R, d \in D, c \in C, p \in P$

$M_{tsdp}, N_{sdp}, L_{tdp}, Y_{rp}, Y_{dp}, X_{rkp}, X_{djp} \in \{0,1\} \qquad \forall\, t \in T, s \in S, d \in D, p \in P$ (29)

Eq. (1) is the objective function of minimizing transportation costs, subassembly's purchases, repairing and opening facility centers. Eq. (2) shows the total number of sub-assemblies bought from suppliers that cannot oversteps those supplier's sub-assemblies capacity in each period. Eq. (3) shows that number of final assembled products should not exceed the disassemblers' capacity. Eq. (4) guarantees that distributed amount of final assembled product from retailers to customers should not exceed the capacity of those retailers. Eq. (5) declares that customer's demand must be completely satisfied. Eq. (6) says the total products carried from collection centers to opened repair and disassembly centers should not overstepped the capacity of collection centers. Eq. (7) shows the total products going out of repair centers to retailers should not exceed the repair centers. Eq. (8) guarantees the total number of sub-assemblies carried from disassembly centers to assembly and dump centers should not overstep the capacity of disassembly centers.

Eq. (9) shows the total sub-assemblies purchased from suppliers and sub-assemblies received from disassemblers opened from previous period, is equal to the total sub-assemblies carried from assemblers to retailers in each period that are assembled in final product's form. Eq. (10) shows that the whole collected products in assemblers and all the repaired products in repair centers opened in the past period are the same as the total products distributed from retailers to costumers in every period. Eq. (11) tells us the costumer's gathered products are in the range of "minimum" to "maximum" collected limits, according to θ_min and θ_max collecting ratios.

Eq. (12) expresses that the equality between λ percent of the total collected products from costumers toward collection centers and entire number of products moved to repair centers. Eq. (13) gives assurance that the total number of repaired products in repair centers is the same as total number of products that are moved from repair centers to retail centers in each period. Eq. (14) shows that the rest of the products collected from costumers via collection centers are carried to disassembly centers in each period. Eq. (15) says that the total useless sub-assemblies detected by disassemblers, is the same as total number of products moved to dump centers. Eq. (16) shows that the total remaining sub-assemblies detected as useful by disassemblers are equal to the total sub-assemblies that must be carried to assemblers. Eq. (17) and (18) guarantee that according to disassembly activity graphs exactly one of the next activity pairs is chosen and preformed from among all the disassemblers in each period. Eq. (19) is assigning the constraint and it guarantees that each disassembly activity in each period is assigned to exactly one work station in every disassembler that is open. Eq. (20) is used to prevent any violation of transposition relationships in each "and/or" graph. Eq. (21) guarantees that the time spent for disassembly of each product in every work station does not violate the cycle time that is the same as division of "one working shift time" by "The total number of assembled sub-assemblies" in each period.

Eq. (22) insures the number of repair centers allocated to retailers will not surpass the potential repair centers sites. Eq. (23) guarantees that the number of disassembly centers assigned to assemblers will not surpass the potential disassembly centers sites. Eq. (24) says that disassembly center d∈D, can serve disassembler j∈J during period p∈P when its corresponding disassembly center is open in the same period. Eq. (25) indicates that the repair center r∈R during the period p∈P, can serve the retailer k∈K when its corresponding repair center opens in that same period. Eq. (26) says that the opened repair center during period p∈P can serve at least one retailer in that period. Eq. (27) shows that the disassembly center opened during period p∈P can serve at least one assembler in that period. And in the end Eqs. (28) and (29) indicate the real and binary variables of the problems. Linearization: To convert nonlinear constraints as Eq. (6)-(10), (12), (14)-(16) to linear ones in this section, assuming that the product of a binary variable and a positive variable are integers, it is illustrated how to linearize Eq. (7) as an example:

$$\sum_{k\in K} E_{rkp}\,.X_{rkp} \qquad \forall\, r\in R, p\in P$$

$$E_{rkp}.X_{rkp} = \begin{cases} E_{rkp} & X_{rkp}=1 \\ 0 & X_{rkp}=0 \end{cases} \tag{30}$$

$$E_{rkp} \le M.X_{rkp} = \begin{cases} E_{rkp} \le M & X_{rkp}=1 \\ 0 & X_{rkp}=0 \end{cases} \tag{31}$$

As the constant M is large enough, variable E_{rkp} will be released when $E_{rkp} \le M$. For linearity, the Eq. (7) should be replaced with the following two constraints:

$$\sum_{k\in K} E_{rkp} \le f_{rp} \qquad \forall\, r\in R, p\in P \tag{32}$$

$$E_{rkp} \le M.X_{rkp} \qquad \forall\, r\in R, p\in P \tag{33}$$

With the explanations presented, the nonlinear constraints of the mathematical model are linearized as follow:

$$\sum_{r\in R} B_{mrp} + \sum_{d\in D} S_{mdp} \le e_{mp} \qquad \forall\, m\in M, p\in P \tag{34}$$

$$\sum_{r\in R} B_{mrp} + \sum_{d\in D} S_{mdp} \le e_{mp} \qquad \forall\, m\in M, p\in P \tag{35}$$

$$F_{dcp} + \sum_{l\in L} Z_{djcp} \le g_{dcp} \qquad \forall\, d\in D, c\in C, p\in P \tag{36}$$

$$F_{dcp} + \sum_{l\in L} Z_{djcp} \le M.X_{djp} \qquad \forall\, d\in D, c\in C, p\in P \tag{37}$$

$$\sum_{i\in I} X_{ijcp} + \sum_{d\in D} Z_{djc(p-1)} = r_c \sum_{k\in K} Y_{jkp} \qquad \forall\, j\in J, c\in C, p\in P \tag{38}$$

$$\sum_{d\in D} Z_{djc(p-1)} \le M.(r_c.\sum_{k\in K} Y_{jkp}) \qquad \forall\, j\in J, c\in C, p\in P \tag{39}$$

$$\sum_{j\in J} Y_{jkp} + \sum_{r\in R} E_{rk(p-1)} = \sum_{l\in L} W_{klp} \qquad \forall\, k\in K, p\in P \tag{40}$$

$$\sum_{r\in R} E_{rk(p-1)} \le M.\ \sum_{l\in L} W_{klp} \qquad \forall\, k\in K, p\in P \tag{41}$$

$$\theta_{min} \sum_{k\in K} W_{klp} \le \sum_{m\in M} A_{lmp} \le \theta_{max} \sum_{k\in K} W_{klp} \qquad \forall l\in L, p\in P \tag{42}$$

$$\lambda \sum_{l\in L} A_{lmp} = \sum_{r\in R} B_{mrp} \qquad \forall m\in M, p\in P \tag{43}$$

$$\sum_{m\in M} B_{mrp} = \sum_{k\in K} E_{rkp} \qquad \forall\, r \in R, p \in P \tag{44}$$

$$(1-\lambda)\sum_{l\in L} A_{lmp} = \sum_{d\in D} S_{mdp} \qquad \forall m \in M, p \in P \tag{45}$$

$$S_{mdp} \le M.Y_{dp} \qquad \forall\, m \in M\,, p \in P \tag{46}$$

$$B_{mrp} \le M.Y_{rp} \qquad \forall\, m \in M\,, p \in P \tag{47}$$

$$r_c(1-\mu)\sum_{m\in M} S_{mdp} = F_{dcp} \qquad \forall\, d \in D, c \in C, p \in P \tag{48}$$

$$r_c.\mu \sum_{m\in M} S_{mdp} = \sum_{j\in J} Z_{djcp} \qquad \forall\, d \in D, c \in C, p \in P \tag{49}$$

$$S_{mdp} \le M.X_{djp} \qquad \forall\, m \in M\,, p \in P \tag{50}$$

The use of above mathematical model can be useful for optimizing the following decisions: 1) Optimizing the number of products that must be delivered from the final customer to the collection centers to keep down the whole purchase costs as much as possible. 2) Optimizing the number of gathered products that would be renovated or disassembled. 3) Designing an optimal DLB structure which keeps the quantity of workstations at minimum and the quantity of products collected at maximum. 4) Optimizing locating and opening disassembly and refurbishing lines due to bring the situation nearer to the real world.

4 Solution of the numerical sample

To express better conception of the proposed model, a general network based on the research by Ozceylan et al. (2014) has been shown in Figure 2. This sample issue consisted of 4 suppliers, 2 assemblers, 2 retailers and 4 customers in forwarding chain and 2 collection centers, 4 potential disassembly centers and a place for removing reversed chain. Suppliers supply different sub montages of a flashlight to send assembly centers. A sample product has been shown with separated subassemblies in Figure 3. at period 1, subassemblies are assembled in assembly center and they are transferred to retailers and customers.

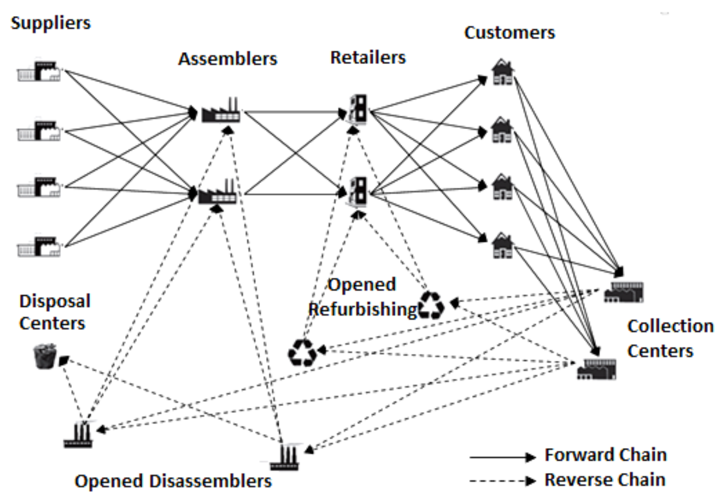
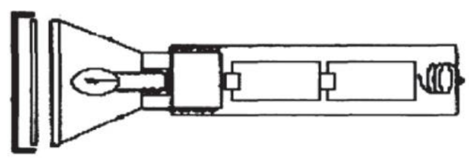
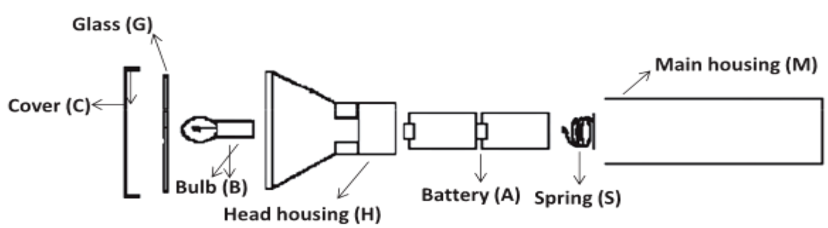


Figure 2. Graph of CLSC network for flashlight



(a)



(b)

Figure 3. (a) flashlight and (b) its subassemblies (Ozceylan, 2014)

The reversed chain flow is initiated by collecting applied flashlights from customers and collecting rate is considered in (Θ_{min}=0.2, Θ_{max}=0.8) interval. Regarding the studies on Sic parameter, $c \in C$ subassembly purchase cost from $i \in I$ supplier directly affected the collection rate in a way that higher Sic value increased collected values of applied products from costumers. In the next reversed chain level (1-λ) % of the collected products opened towards disassemblers and λ% that will not require fundamental process anymore will be transferred toward opened restructured centers. Then, manual flashlights are transferred for retailers allocated to it from opened restructure centers during next stages. The presence of active disassemblers is necessary for transforming return products to their subassemblies and must be balanced. After disassembly operation, subassemblies are categorized to general classes. The first category is the usable subassemblies transferred to disassemblers in the next period and the second category is sub assembles directed to removal center. It is assumed that the number of output sub assembles from disassembly centers is an integer number of complete products. The product which flows in this chain has the ability to be disassembled in 3 forms presented as a graph and/or in the Figures 4 and 5 and any letter indicates its corresponding subassembly. According to above explanations the presence of the artificial nodes along with the normal ones is necessary. Figure 4 indicates that the flashlight (product) can be disassembled from path 1 or 2. As it is shown in this figure, three complete disassemblies (2-5-7-8-9-10), (2-4-6-7-8-9-10) and (1-3-6-7-9-10) are available. In Figure 5, graph and/or transformed has been done for better relation between transpositions. Based on Figure 5 there are eight artificial nodes (Aa, a∈ [0, 1, 2, 3, 4, 5, 6, 7]) and 10 normal nodes (Bt, t= [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]). The time related to normal nodes is inserted above it and the artificial nodes time is zero. Transportation unit cost equals with 5.23 money unit per ton/kilometer with trailer vehicles.

Workstation opening fixed cost in any disassembly line per each cycle (Odp) is considered as 1000 money unit. Maximum number of workstations in any disassembly line per any period (Sdp) was 6 cases. This sample issue is designed during two periods and any period consisted of 4 months (16 weeks) with 6 workdays in a week and 9 hours a day. Also, the total accessible time is considered 60000 Unit/time. Further information about the distances, capacities and demands for this numerical sample has been explained in (Ozceylan, 2014).

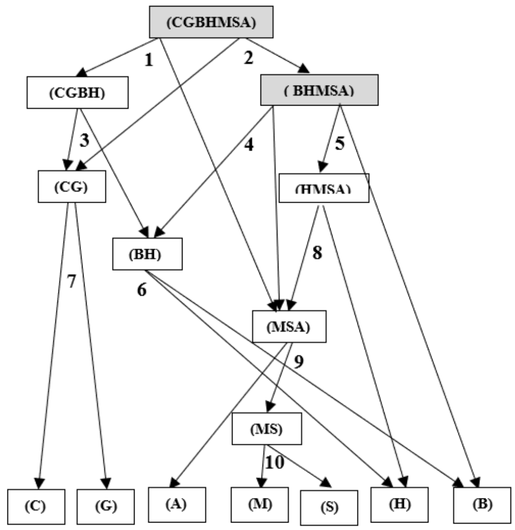


Figure 4. Disassembly graph

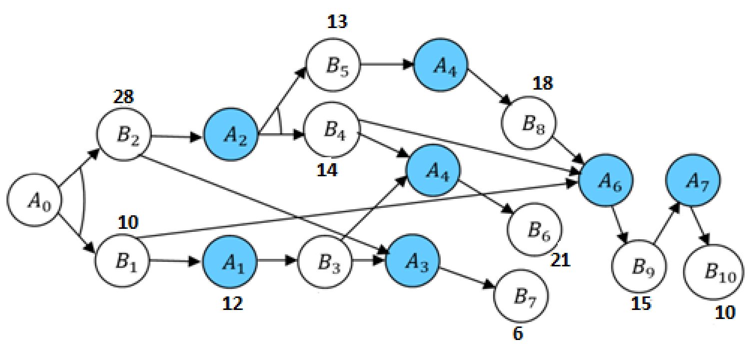


Figure 5. Flashlight assembly transformation graph

The computational process has been conducted by GAMS software with CPLEX solution engine. Figures 6 and 7 show optimal distribution plans for closed-loop supply chain and optimal disassembly line balancing in the first period respectively. Lines marked with stars in these figures are fully described in Tables 1 and 2.

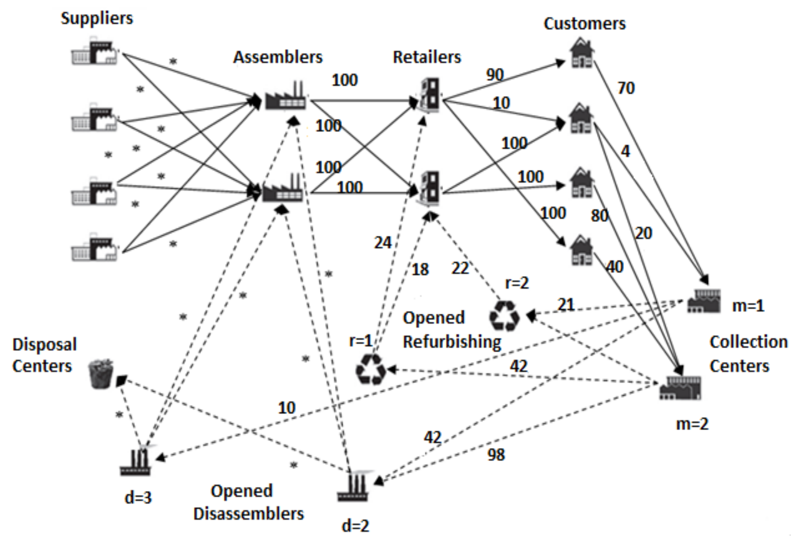


Figure 6. Optimal distribution plan for CLSC in the first period

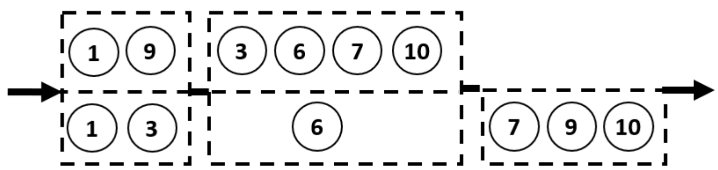


Figure 7. Optimal disassembly line balancing in the first period

Table 1. Subassemblies delivered to the assemblers and disposal centers in the first period

Subassembly	1	2	3	4	5	6	7
Supplier 1							
assembler 1	100	100	100	100	100	100	100
assembler 2	-	-	-	-	-	-	100
Supplier 2							
assembler 1	100	100	100	100	100	100	100
assembler 2	100	100	100	100	100	100	100
Supplier 3							
assembler 1	-	-	-	-	-	-	100
assembler 2	100	100	100	100	100	100	100
Supplier 4							
assembler 1	-	-	-	-	-	-	100
assembler 2	-	-	-	-	-	-	100
disassembly center 2							
assembler 1	1	1	1	1	1	1	96
assembler 2	97	97	97	97	97	97	100
disposal center	42	42	42	42	42	42	84
disassembly center 3							
assembler 1	4	4	4	4	4	4	13
assembler 2	3	3	3	3	3	3	1
disposal center	3	3	3	3	3	3	6

Table 2. Optimal disassembly line balancing for disassemblers 1 and 2 in the first period

	Workstation	**Tasks assigned**	**Workstation time**	**Idle time**
Disassembler 1	1	1-9	25	32.86
	2	3-6-7-10	49	8.86
	Total			**41.72**
Disassembler 2	1	1-3	22	12.29
	2	6	21	13.29
	3	7-9-10	31	3.29
	Total			**28.78**

Figures 8 and 9 show optimal distribution plans for CLSC and optimal DLB in the second period respectively. The objective function's optimal amount is $ 1,380,784 for a period of 32.68 seconds. Lines marked with stars in these figures are fully described in Tables 3 and 4.

Table 3. Subassemblies delivered to the assemblers and disposal centers during the second period

Subassembly	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Supplier 1							
assembler 1	100	100	100	100	100	100	100
assembler 2	-	-	-	-	-	-	99
Supplier 2							
assembler 1	-	-	-	-	-	-	-
assembler 2	100	100	100	100	100	100	100
Supplier 3							
assembler 1	-	-	-	-	-	-	83
assembler 2	100	100	100	100	100	100	100
Supplier 4							
assembler 1	-	-	-	-	-	-	-
assembler 2	-	-	-	-	-	-	-
disassembly center 2							
assembler 1	-	-	-	-	-	-	-
assembler 2	42	42	42	42	42	42	84
disposal center	18	18	18	18	18	18	36
disassembly center 3							
assembler 1	-	-	-	-	-	-	-
assembler 2	7	7	7	7	7	7	3
disposal center	3	3	3	3	3	3	6

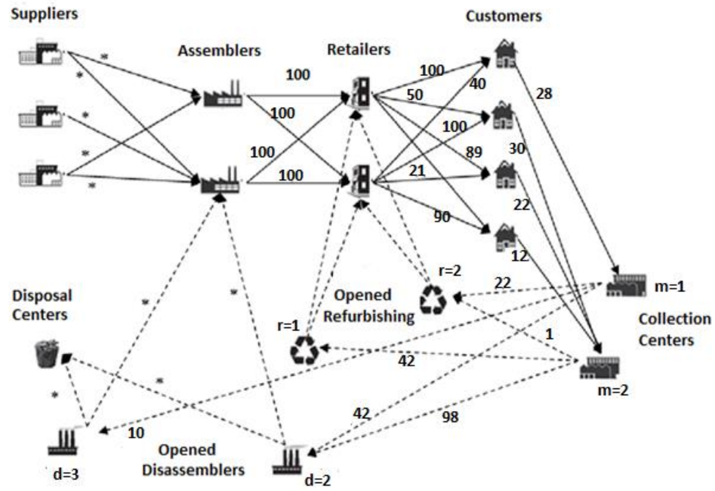


Figure 8. Optimal distribution plan for CLSC in period 2

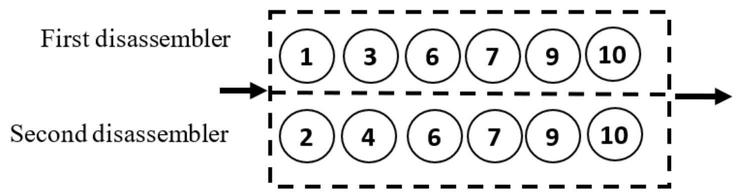


Figure 9. Optimal disassembly line balancing in period 2

Table 4. Optimal disassembly line balancing for disassemblers 1 and 2 during the second period

	Workstation	**Tasks assigned**	**Workstation time**	**Idle time**
Disassembler 1	1	1-3-6-7-9-10	74	0
	Total			**0**
Disassembler 2	1	2-4-6-7-9-10	94	495.1
	Total			**495.1**

5 Examination the proposed model

5.1 Solution based GAMS software

In this section, some small-scale samples have been randomly generated. The proposed model is programmed in GAMS optimizing modeling Software version 24.1.2. Then typical problems are done by the CPLEX solving engine and the results are achieved according to Table 5. In all generated samples, a maximum of 6 workstations are located in disassembly lines. Also the number of potential refurbishing and disassembly centers are 4 sites. Furthermore the number of activities is 10 in all samples. The solution time mentioned in Table 5 is the time for minimizing the costs as the objective function. As it is shown in this table and in Figure 10, the problem solution time increases exponentially by increasing samples' dimensions which is reasonable with regard to the NP-hardness of the problem of facility location in the CLSC integration model and DLB. The increasing time expresses the fact that GAMS software is not time efficient at wide-ranging subjects. So the development of a metaheuristic algorithm is essential for solving these kinds of problems at a reasonable time.

Table 5. Solution time of some sample problems in GAMS software

Sets	**Sample 1**	**Sample 2**	**Sample 3**	**Sample 4**	**Sample 5**
Suppliers (i)	2	2	4	5	5
Assemblers (j)	2	2	2	4	3
Retailers (k)	2	2	2	3	3
Customers (l)	2	2	4	4	4
Collection Centers (m)	2	2	2	2	2
Furbishing Centers (r)	3	4	4	4	4
Disassembly Centers (d)	3	4	4	4	4
Solution Time(s)	**1.98**	**3.36**	**32.68**	**176.91**	**312.83**

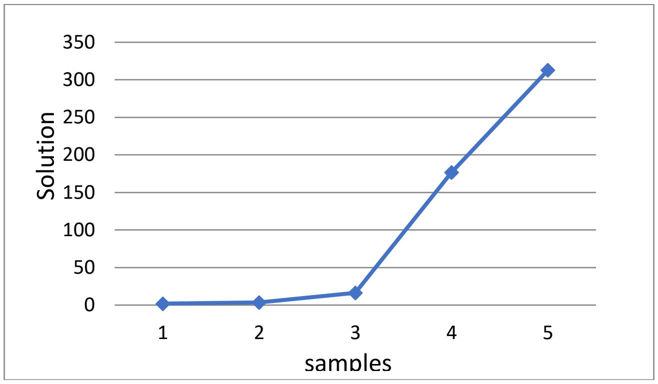


Figure 10. Increased solution time in GAMS by increasing samples 'dimensions

The largest sample problem solved by GAMS software consist of 6 suppliers, 5 assemblers, 4 retailers, 6 customers, 4 collection centers, 4 refurbishing centers and 4 disassembly centers optimal objective function's value of which is $ 395,0857.04 obtained up to 7200 seconds for this problem. The computational results show that the locating of potential facilities for refurbishing and disassembly and the optimal opening of the facility that is based on the distance minimization approach, and finally keeping the entire cost of the CLSC and DLB at minimum is closer to the real world compared to the case that there were no potential sites. Also further attention to the distance factor has resulted in more valid and optimal results through applying the proposed mathematical model. On the other hand, it was observed that by increasing the dimensions of the problem, the problem solving time expands exponentially. Therefore, with the aim of gain optimal solutions in a reasonable time, an innovative or metaheuristic algorithm is needed. The next section focuses on the design of a metaheuristic algorithm for solving the research problem.

5-2 Solution based genetic algorithm

Considering the NP-HARD nature of the problem and growth of its solving time in large dimensions in GAMS optimization software, a genetic metaheuristic algorithm has been developed. Using metaheuristic methods has grown significantly to achieve satisfactory solutions in combinatorial optimization problems. Because of problems in bringing conditions to the real world situations, increasing the complexity of them and the failure of the mathematical models to find optimal points, metaheuristic algorithms have been considered greatly. These algorithms are inspired by nature that is used in optimization problems. Unlike the exact methods, these methodologists are applicable large scale issues and provide suitable solutions in a reasonable time. In this section, the details of the solution algorithm, including the method of representing the problem, the production of the initial solution and the neighboring structures of the problem, are described.

Genetic Algorithm: metaheuristic algorithms are divided into two groups: the algorithms which are based on population and based on local search. In population based algorithms, a population of solutions is generated and they will be updated during search processes. In the search based ones, a unique answer is generated as the initial solution and all updates are applied to this solution. Due to the well performance of population-based algorithms, the Genetic Algorithm is applied as one of the most popular algorithms in this type of metaheuristic methods. Genetic algorithm (GA) is a search base technique in artificial intelligence algorithms bunch which find approximate solution for optimization problems. It is a subset of evolutionary algorithms and uses biological methods like inheritance and mutation. It is considered as a global search technique as it imitates the laws of natural biological evolution. GA applies the best survival law, hoping for better answers. In each generation, better approximations are obtained by the selection process corresponding to the value of the solutions and reproduction of the selected solutions with the help of operators that are patterns of natural genetics. This process makes the new generations more consistent with the problem.

GA starts with a set of solutions that are called the main population and it is shown through a sequence of chromosomes. In fact, each chromosome represents a person in the population. Then from this population, the parents are selected and the children are produced by crossover and mutation operators. After children production, a new population emerges consisting of children and the main population. Among the members of the new population, those who are better are selected as the main population in the next generation. This procedure is repeated until the optimal solution is achieved.Modeling: there are many real and binary variables in the current research so in order to decrease problem solving time and to increase its performance each chromosome corresponds to a variable and as a vector. Also each gene is according to the total indices of related variable. The amount of each parts and each chromosome size depends on the criteria like type of variable, number and amount of its indices. For example consider variableY_{jk1}, which is a positive integer variable with two assemblers and two retailers. First, a random number in range of (0,1) is given to each of the genes (specific variables) and since the total percentages of first and second elements isn't equal to one, normalization will be done (Figure 11).

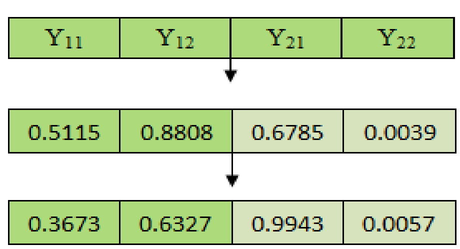


Figure 11. Representing solutions for continuous variables (full product)

The above vector chromosome is composed of two parts: the first part corresponds to the first assembler and the second part corresponds to the one.

Each chromosome's size corresponding to the number of the retailers is 2 and 4 as a whole. Another example is X_{ijc1} or X_{ijc} that indicates the number of C type sub-assemblies moved between i-th and j-th assembler and is responsible for supplying subassemblies 1, 2 and 7 (Figure 12).

X_{111}	X_{112}	X_{117}	X_{121}	X_{122}	X_{127}	X_{211}	X_{212}	X_{217}	X_{221}	X_{222}	X_{227}

Figure 12. Representing solutions for continuous variables (subassembly)

As seen in Figure 12, X_{ijc} the above chromosome is made of 4 parts and each part has 3 variables or genes, so that its overall size is 12. Here each one of elements (genes) is given a random value between zero and one and finally normalization will be done according to each part of the chromosome. Representation of binary variables is different since it has to deal with opening or not opening of disassembly repair centers. For example consider y_{r1} or y_r binary variable with assumption of 4 potential repair center. For instance, in Figure 13, zero and one numbers are randomly given to each element which shows opening the first and fourth repair centers and not opening the second and third ones.

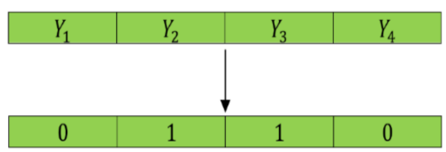


Figure 13. Representing solutions for binary variables

Generally, in order to produce a population, variables are randomly initialized first. They are converted to natural numbers acceptable for the problem while constraints of the problem are applied in the same time. Then a main chromosome string is produced by gathering chromosomes that each one represents a variable and in the end the objective function of each population is calculated. In fact, variables in the algorithm are mapped to understandable variables of the problem. Crossover operator: crossover is one of population producing methods that its execution in research problem is different depending on the type of variable. There are a series of none-negative binary and real variables in mathematical model of this problem that crossover operator might be applied on only some of these variables. Parents' selection policy in problem solving algorithm is roulette wheel and premier parents have a higher chance of being selected. After selecting two parents using roulette wheel, a random number (Q) is created, and accordingly the number of variables to be chosen for crossover or mutation operations will be selected. Asynchronous execution of crossover or mutation operation on all decision variables is due to not using the random search. The general procedure of the mentioned operation is based on generating thebinary numbers for establishing or opening the repair and disassembly centers. Then numbers are generated in continuous range between zero and one in a completely randomized manner for all existing matrixes containing real decision variable amounts. For real and positive variables like $\mathrm{Y_{jk1}}$ the crossover operation first creates a $\propto$ matrix with the same dimensions as $\mathrm{Y_{jk1}}$. So that values of its elements are within the $[-\gamma, 1+\gamma]$ range. Gamma γ is a small number around 0.1 that provides the possibility of overtaking the parents for children. Then new children are generated according to Eqs. 50 and 51:

$$y_1 = \propto X_1 + (1 - \propto) X_2 \tag{51}$$

$$y_2 = (1 - \propto) X_1 + \propto X_2 \tag{52}$$

Where, $\mathrm{X_1}$ and $\mathrm{X_2}$ are the parents and it indicates that it inherits as much as $\propto$ from parent one and $1-\propto$ from parents two. For example crossover sub-operation for $\mathrm{Y_{jk1}}$ parents is as Figure 14:

0.5115	0.8808	0.6785	0.0039	First parent
0.8287	0.4690	0.6859	0.0873	Second parent

Figure 14. Sample parents for continuous variable

Creating an $\propto$ matrix of random numbers in [-0.1, 0.1] range and also generated children proportionate with $\mathrm{Y_{jk1}}$ is as Figure 15:

$\boldsymbol{\alpha}$ =	0.6459	0.5729	0.8188	0.3652	
	0.6238	0.8416	0.6798	0.0568	First child
	0.7164	0.6449	0.6846	0.0344	Second child

Figure 15. Children generated by remarked crossover operator

Also one of single point, double point or uniform crossovers will be used randomly for each binary variable. For example, $\mathrm{Y_{r1}}$ binary variable with four potential repair centers is considered. In the beginning thr first and second child are generated based on roulette wheel, as Figure 16:

Y_1	Y_2	Y_3	Y_4

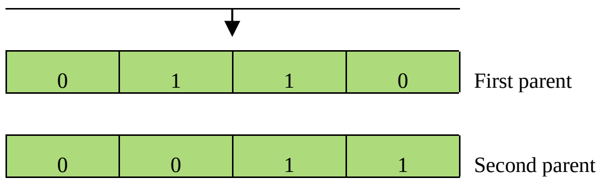


Figure 16. Sample parents for binary variable

Then the following children are generated from parent chromosomes by selecting single point cut crossover method (Figure 17).

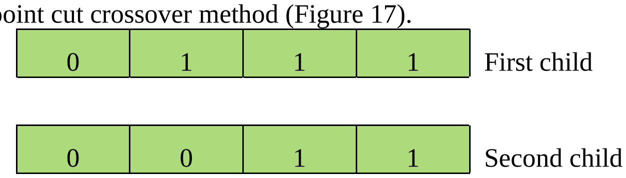


Figure 17. Children generated by one-point crossover (binary variable)

Mutation operator: similar to crossover operator, mutation operator is also randomly applied on a number of variables. In this operator, a change is made in variables' value so that they have a completely different value than their initial one. A different mutation operator is applied on natural and binary variables here as well. First a parent is randomly chosen. Mutation for a binary variable like Y_{r1} is as follows, that at first according to the equation below, it is the number of variables to be changes is determined. Since the values are zero and one, it just changes the zeros to one and vice versa. For example:

$m = \max\{\pi_m \times n, 1\}$, $\pi_m = (5 - 20)\%$, $n = number\ of\ elements$

$y_{r1} = [1 \quad 0 \quad 0 \quad 1]$, $m = max[0.2 \times\ 4, 1] = 1 \quad \rightarrow \quad (random\ selection\ of\ one\ of\ genes)$

$\acute{y}_{r1} = [1 \quad 0 \quad 0 \quad 1]$

For natural variables first according to the equation above, number of elements that must be changed is randomly chosen and then the elements are randomly selected, and selected elements are changed.

$x^{\text{new}} = x^{old} + \delta\, N(0, 1); \;\; \delta = 0.1(x_{max} - x_{min})$

For example consider the variable Y_{jk} (indicating number of products moved between assembler j and retailer k):

$$\acute{y}_{jk1} = \begin{bmatrix} 0.10 & 0.56 & 0.28 & 0.84 \\ 0.65 & 0.99 & 0.05 & 0.77 \\ 0.87 & 0.25 & 0.78 & 0.65 \\ 0.23 & 0.28 & 0.11 & 0.77 \end{bmatrix}$$

, $max = 1$, $min = 0 \delta = 0.1$

So mutation is applied on one of the elements for instance element (2, 3) with N(0,1) = - 0.6 assumption:

$$x^{\text{new}} = x^{\text{old}} + \delta\, N(0,1) = 0.25 + 0.1\,(-0.6) = 0.19$$

Therefore

$$\acute{y}_{jk1} = \begin{bmatrix} 0.10 & 0.56 & 0.28 & 0.84 \\ 0.65 & 0.99 & 0.05 & 0.77 \\ 0.87 & 0.19 & 0.78 & 0.65 \\ 0.23 & 0.28 & 0.11 & 0.77 \end{bmatrix}$$

Figure 18 shows genetic algorithm flowchart for developing the model.

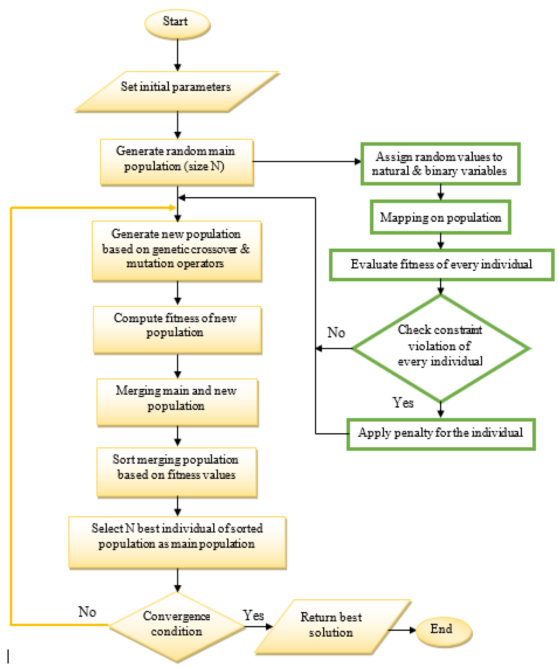


Figure 18. Flowchart of the solution by GA

In this part a few problems in various scales are made randomly. For accurate solving of the problems, the proposed method is programmed in GAMS. Then some instance problems are performed by CPLEX using a system with Core i5, 2.5 Giga Hertz processor and a 4 Giga Byte memory. The most permitted time for GAMS software is 7200 seconds. The proposed solving algorithm is also programmed in MATLAB 2015. In table 6, error is calculated by Eq. 53 that f^{best} shows the best value acquired after 10 times of executing the algorithm and f^{GAMS} indicates the solution given by GAMS as the optimal solution or the best proposed solution.

$$\frac{f^{best} - f^{GAMS}}{f^{GAMS}} \times 100 \tag{53}$$

In Table 6, the error was not calculated in cases that algorithm's given solution was better than the best solution proposed by GAMS optimization software, and it is shown in the table with dashes. Bold values in the column concerning the best answer after 10 times of executing the algorithm show that in large size problems, the proposed algorithm presented better answers than the local optimal answers given by GAMS optimization algorithm on a 7200 seconds time limit.

Table 6. Numerical experiments and calculations for evaluation of algorithms' Performance

Error rate	Proposed run time	Proposed Algorithm results in 10 runs		GAMS run time	GAMS results	Disassembly Centers	Refurbishing Centers	Collection Centers	Customers	Retailer Centers	Assembly Centers	Suppliers	Sample
		Best	Median										
0.00	28.79	824870	834483	1.98	*824780	3	3	2	2	2	2	2	1
0.00	26.43	**814416**	827205	2.18	*814410	3	3	2	2	2	3	2	2
0.00	29.26	**781270**	795805	1.77	*781202	3	3	2	2	3	2	2	3
0.00	31.11	**829955**	836267	3.59	*829936	4	3	2	2	2	2	3	4
0.00	30.2	**833799**	841013	4.56	*833785	4	4	2	2	2	2	3	5
0.00	31.58	**793906**	809906	6.21	*793835	4	4	2	2	2	3	2	6
0.04	33.21	**985713**	1006325	4.86	*985325	4	4	2	2	3	2	2	7
0.00	28.73	**752950**	809655	1.34	*752867	3	4	2	2	2	2	2	8
0.00	30.43	**596065**	678534	3.36	*595983	4	4	2	2	2	2	2	9
0.00	34.65	**1024565**	1105899	6.52	*1024551	4	4	2	3	2	2	2	10
2.98	34.15	**1421973**	1516947	32.68	*1380784	4	4	2	4	2	2	4	11
0.00	35.74	**1209647**	1467942	6.62	*1209576	4	4	2	4	3	2	4	12
0.00	39.88	**1303569**	1314825	257.41	*1303547	4	4	3	4	2	3	4	13
2.92	48.12	**1361808**	1466265	176.91	*1402871	4	4	2	4	3	4	5	14
0.09	45.25	**1455967**	1543491	312.83	*1457391	4	4	2	4	3	3	5	15
-	64.37	**1596173**	1678037	848.5	**1600123	4	4	3	6	2	2	6	16
-	58.32	**2162625**	2174256	7200	**2362856	4	4	2	6	4	4	6	17
-	83.96	**2569097**	2578444	7200	**2877110	4	4	3	6	4	4	6	18
-	109.21	**2104362**	2128444	7200	**2172110	4	4	3	6	4	5	6	19
-	428.13	**2547976**	2560655	7200	**3957857	4	4	4	6	4	5	6	20

* global optimum achieved by GAMS	** best result achieved in 7200 second

Evolutionary charts presented in Figures 19-22 demonstrate convergence to the best answer that considering the convergence term of proposed algorithm and number of algorithm's solution iterations, the horizontal trail shown in the tailing of evolutionary charts shows this fact in detail.

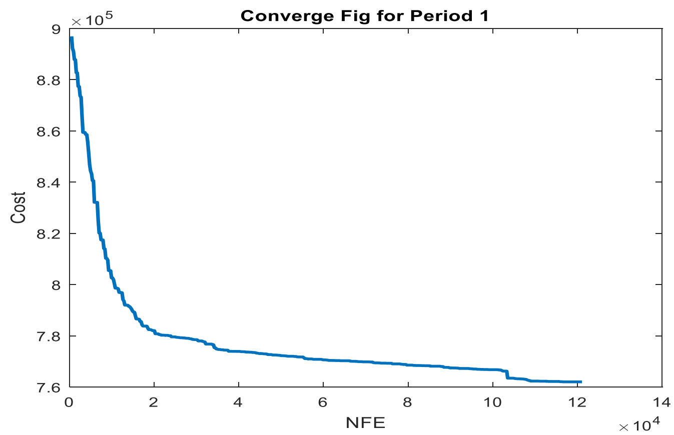


Figure 19. Sample 11's convergence graph (average) in the first period

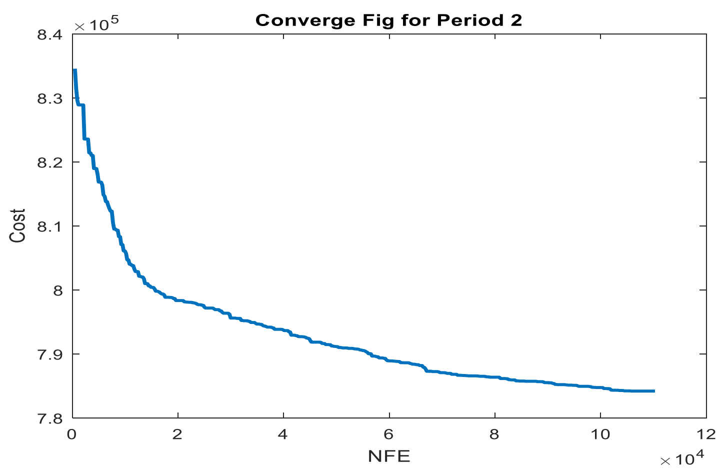


Figure 20. Sample 11's convergence graph (average) in the second period

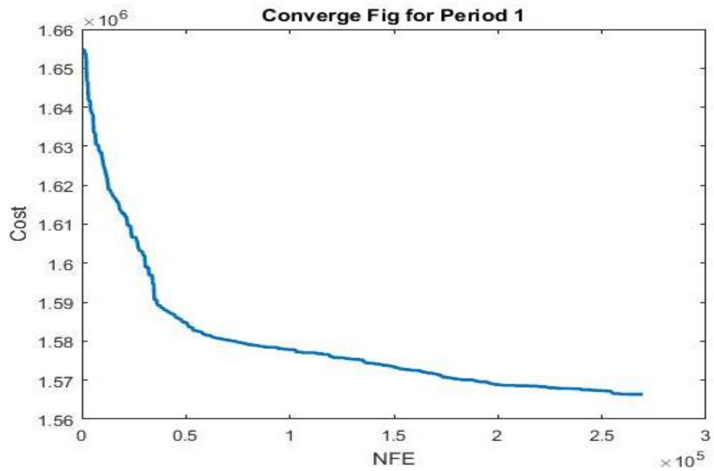


Figure 21. Sample 18's convergence graph (average) in the first period

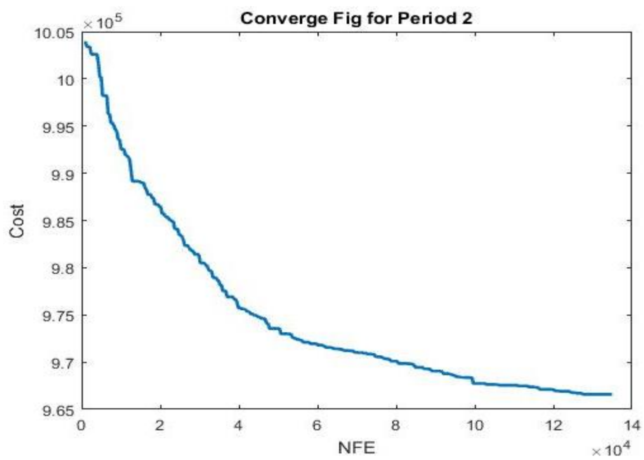


Figure 22. Sample 18's convergence graph (average) in the second period

Based on the results in end column of Table 6 and Figure 23, the suggested solution algorithm has a fair efficiency and its error percentage is trivial compared to the best proposed answer by algorithm after 10 iterations and GAMS's proposed answer. In instances with bigger sizes the proposed algorithm gave a better answer than GAMS optimization software in most cases. Another point is that as shown in Figure 24, even in problems with smaller sizes GAMS optimization software quickly gets ahead of the proposed solution algorithm in terms of solving time which is significantly apparent in Table 6 in problems with larger sizes.

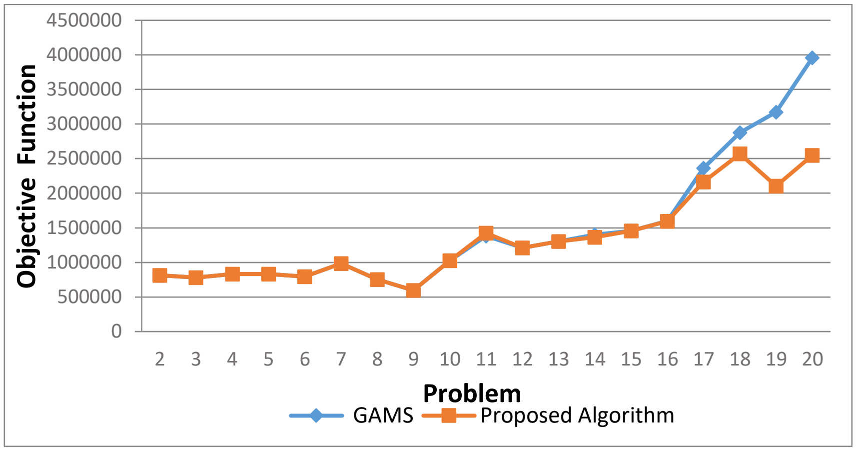


Figure 23. Comparison of objective function values between GAMS and the proposed Algorithm

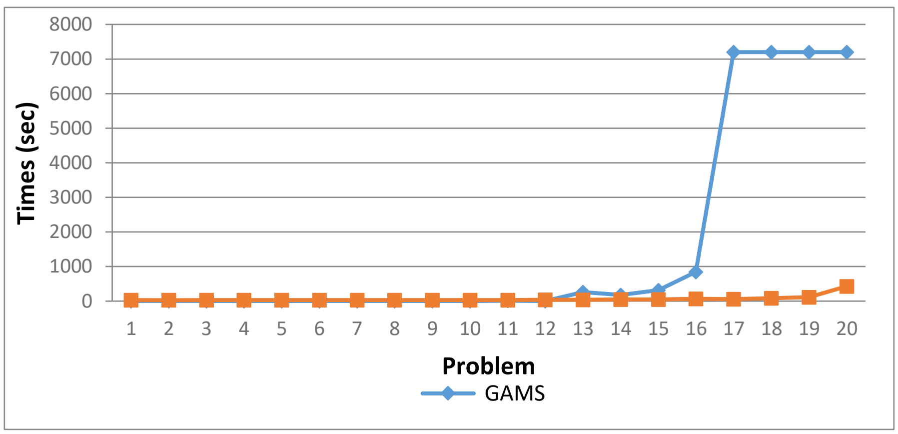


Figure 24. Comparison of required Time between GAMS and proposed Algorithm

Conclusions

Deciding about facility locations has a sensitive affect in closed-loop supply chain networks design. Utility and application of location especially in logistics will never be doubted and is one of the main logistic decisions in supply chain scope. Because of the notable consideration given to the closed-loop supply chain facility location problem, lack of attention to this subject in disassembly line balance in closed-loop supply chain creates a space for further research. The presented problem is important since it works on location of facilities in reverse chain in an integrated model of closed-loop supply chain network design. Proposing a model that can concurrently optimize work stations number in disassembly line, cycle time, total costs, and amount of sub-assemblies bought from suppliers, amounts of sub-assemblies or final products carried between facilities is considered as the innovation share of this research. In this problem it is assumed that there are some of potential sites for choosing repair and disassemble facilities in backward supply chain and on the other hand disassemble lines will be balanced by calculating cycle time, idle time and so on. This problem was modeled as a linear mixed integer mathematical model for minimizing the transport total costs, subassemblies' purchases, repair and opening facility centers. The proposed model can be accurately solved for problems with small sizes. However due to being NP-hard, accurate solution of mathematical model for problems with large sizes is not possible and therefore a solution pursuant to genetic algorithm was proposed to find a solution for the problem. With the aim of measuring the efficiency of the designed model, a number of instance problems were proposed and executed 10 times by the proposed algorithm and then compared with the results of accurate method. Experimental results of numerical examples show that genetic algorithm can acquire near optimal results in a very shorter time than the accurate method. Since the research issue was single objective and includes minimizing the total costs, studying other objective functions in a concurrent manner like maximizing costumers' satisfaction could be an interesting field for future studies. The other serious assumption of the current work is the single product flow in supply chain that should be completely disassembled after collection if necessary. Designing a supply chain by considering multi product flow can be a step to develop the research. Considering other cost and operational elements like choosing transport methods or how to disassemble the products in a closed-loop supply chain flow in a way that product is disassembled in several methods or approaches, brings the research problem similar to the real world.

References

[1] Wang, H.F., & Hsu, H.W. (2010). A closed-loop logistic model with a spanning-tree based genetic algorithm. Computers & Operations Research, 37, 376–389.

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

[3] Altekin, F.T., Kandiller, L. & Ozdemirel, N.E. (2008). Profit-oriented disassembly-line balancing. International Journal of Production Research, 46, 2675-2693.

[4] Gungor, A. & Gupta, S.M. (2002). Disassembly line in product recovery. International Journal of Production Research, 40, 2569–2589.

[5] Ilgın, M.A. & Gupta, S.M. (2010). Environmentally conscious manufacturing and product recovery: a review or the state of the art. Journal of Environmental Management, 91, 563–591.

[6] Ding, L.P., Feng, Y.X., Tan, J.R. & Gao, Y.C. (2010). A new multi-objective ant colony algorithm for solving the disassembly line balancing problem. International Journal of Advanced Manufacturing Technology, 48, 761–771.

[7] Fleischmann, M., Beullens, P., Bloemhof-Ruwaard, J.M. & Van Wassenhove, L.N. (2001). The impact of product recovery on logistics network design. Production and Operations Management, 10, 156–173.

[8] Fleischmann, M., Bloemhof-Ruwaard, J.M., Beullens, P. & Dekker, R. (2004). Reverse logistics network design. Reverse logistics, Springer, Berlin.

[9] Ozceylan, E., Paksoy, T. & Bektaş, T. (2014). Modeling and Optimizing the Integrated Problem of Closed-loop Supply Chain Network Design and Disassembly Line Balancing. Transportation Research Part E: Logistics and Transportation Review, 61, 142–164.

[10] Melo, M.T., Nickel, S., Saldanha-da-Gama, F. (2009). Facility location and supply chain management– A review, European Journal of Operational Research, 196, 401–412.

[11] Hassanzadeh Amin, S. & Baki, F. (2013). A facility location model for global closed-loop supply chain network design, Applied Mathematical Modeling, 37, 4165-4176.

[12] Ghadge, A., Yang, Q., Caldwell, N., Caldwell, N., Konig, K. & Tiwari, M.K. (2016). Facility location for a closed-loop distribution network: a hybrid approach. International Journal of Retail & Distribution Management, 44, 884-902.

[13] Agrawal, S. & Tiwari, M.K., (2008). A collaborative ant colony algorithm to stochastic mixed-model U-shaped disassembly line balancing and sequencing problem. International Journal of Production Research, 46, 1405–1429

[14] Altekin, F.T., & Akkan, C.)2012(. Task-failure-driven rebalancing of disassembly lines. International Journal of Production Research, 50, 4955–4976.

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

[16] Manzini, R. & Bindi, F.)2009(. Strategic design and operational management optimization of a multi stage physical distribution system. Transportation Research Part E, 45, 915–936.

[17] Miranda, P.A. & Garrido, R.A.)2004(. Incorporating inventory control decisions into a strategic distribution network design model with stochasticdemand. Transportation Research Part E, 40, 183-207.

[18] McGovern, S.M. & Gupta, S.M. (2007). Combinatorial optimization analyses of the unary NP-complete disassembly line balancing problem. International Journal of Production Research, 45, 4485–4511.

[19] Ozceylan, E. & Paksoy, T. (2013). Reverse supply chain optimization with disassembly line balancing. International Journal of Production Research, 51, 5985-6001.

[20] Salema, M.I.G., Povoa, A.P.B. & Novais, A.Q.)2009(. A strategic and tactical model for closed-loop supply chains. OR Spectrum, 31, 573–599.

[21] Krikke, H., Hofenk, D. & Wang, Y. (2013). Revealing an invisible giant: a comprehensive survey into return practices within original (closed-loop) supply chains. Resources, Conservation and Recycling, 73, 239–250.

[22] Stindt, D. & Sahamie, R. (2014). Review of research on closed loop supply chain management in the process industry, Flexible Services and Manufacturing Journal,26, 268–293.

[23] Fleischmann, M., Bloemhof-Ruwarrd, J.M., Dekker, R., DerLann, E., Nunen, J.A.E.E. & Wassenhove, L.N. (1997). Quantitative models for reverse logistics: a review, European Journal of Operational Research, 103, 1–17.

[24] Daniel, V., Guide, R., Luk, Jr. & Van Wassenhove, N. (2009). The evolution of closed-loop supply chain research, Operations Research, 57, 10–18.

[25] Akcali, E., Cetinkaya, S. (2011). Quantitative models for inventory and production planning in closed-loop supply chains, International Journal of Production Research, 49, 2373–2407.

[26] Govindan, K., Popiuc, M.N. & Diabat, A. (2013). Overview of coordination contracts within forward and reverse supply chains, Journal of Cleaner Production, 47, 319–334.

[27] Kaya, O., Urek, B. (2015) A Mixed Integer Nonlinear Programming Model and Heuristic Solutions for Location, Inventory and Pricing Decisions in a Closed Loop Supply Chain, Computers & Operations Research, 93-103.

[28] Saman Hassanzadeh, A., Baki, F. (2016) A facility location model for global closed-loop supply chain network design, Applied Mathematical Modelling, 41, 316-330.

[29] Al-Salem, M., Diabat, A., Dalalah, D. & Alrefaei, M,. (2016). A closed-loop supply chain management problem: Reformulation and piecewise linearization, Journal of Manufacturing Systems, 40, 1-8.

[30] Xu, Z., Pokharel, S., Elomri, A. & Mutlu, F. (2016). Emission policies and their analysis for the design of hybrid and dedicated closed-loop supply chains, Journal of Cleaner Production, 142, 4152-4168.

[31] Jabbarzadeh, A., Haughton, M., Khosrojerdi, A. (2017). Closed-loop Supply Chain Network Design under Disruption Risks: A Robust Approach with Real World Application, Computers & Industrial Engineering, 116, 178-191.

[32] Bhattacharyya, S., Choudhary, A., Shanka, R,. (2017). A decision model for a strategic closed-loop supply chain to reclaim End-of-Life Vehicles, 195, 273-286.

[33] Paydar, M., Babaveisi, V., Safaei, A. (2017). An engine oil closed-loop supply chain design considering collection risk, Computers and Chemical Engineering, 104, 38-55.

[34] Ahmadzadeh, E., Vahdani, B. (2017). A location-inventory-pricing model in a closed loop supply chain network with correlated demands and shortages under a periodic review system, Computers & Chemical Engineering, 101, 145-166.

[35] Soleimani, H., Govindan, K., Saghafi, H. & Jafari, H. (2017). Fuzzy Multi-Objective Sustainable and Green Closed-Loop Supply Chain Network Design, Computers & Industrial Engineering, 109, 191-203.

[36] Wu, GH., Hsu, LM., Chang, CK. (2018). Comparisons of Interactive Fuzzy Programming Approaches for Closed-loop Supply Chain Network Design under Uncertainty, Computers & Industrial Engineering, 125, 500-513.

[37] Cheraghalipour, A., Paydar, M., Hajiaghaei –Keshteli, M. (2018). A Bi-objective Optimization for Citrus Closed-Loop Supply Chain Using Pareto-Based Algorithms, 69, 33-59.

[38] Jerbia,R., Boujelben,MK., Sehli, MA., Jemai, Z. (2018). A stochastic closed-loop supply chain network design problem with multiple recovery options, Computers & Industrial Engineering, 118, 23-32.

[39] Vahdani, B., Ahmadzadeh, E. (2019). Designing a realistic ICT closed loop supply chain network with integrated decisions under uncertain demand and lead time, Knowledge-Based Systems 179, 34-54.

[40] Hajipour, V., Tavana, M., Di Caprio, D., Akhgar, M., Jabbari, Y. (2019). An optimization model for traceable closed-loop supply chain networks, Applied Mathematical Modelling, 71, 673-699.

[41] Zhen, L., Huang, L., Wang, W. (2019). Green and Sustainable Closed-Loop Supply Chain Network Design under Uncertainty, Journal of Cleaner Production, 227, 1195-1209.

[42] Yavari, M., Geraeli, M. (2019). Heuristic Method for Robust Optimization Model for Green Closed-Loop Supply Chain Network Design of Perishable Goods, Journal of Cleaner Production, 226, 282-305.