Informatics Bulletin, Faculty of Computers and Artificial Intelligence, Helwan University
Published Online ***Vol 4 Issue 2, July 2022***
(https://fcihib.journals.ekb.eg)

A Hybrid Cellular Genetic Algorithm-Based Mixed Variable Optimization Model for Hospital Waste Closed Loop Supply Chain Reverse Logistics Networks

Alaa Fahim
Mathematics Departement,
Faculty of Science,
Assiut university
alaa@aun.edu.eg

Esraa Ramadan
Mathematics Departement,
Faculty of Science,
Assiut university

Abstract:

Design of hospitals waste supply chain network considering both forward and reverse flows has been increasing day by day due to resource constraints, increased costs and the importance of utilizing returned waste. This paper attempts to integrate both forward and reverse logistics to design a general Closed Loop Supply Chain Reverse Logistics (CLSCRL) network. In this paper, a Mixed Variable Optimization (MVO) model is created for minimizing costs of hospital waste CLSCRL networks. The total costs for CLSCRL optimize cost of opening the gathering stations, treating stations, distribution stations and received hospitals, shipped cost and processing cost. Finally, a schedule is proposed of flows of hospital waste in the network. A hybrid Cellular Genetic Algorithm (CGA) is used for solving the MVO model to be (CGAMV). Pattern Search Method (PSM) is added to the proposed method, to be (CGAMV ^{P}) to make more intensification on the best solutions in proposed grid. The grid structure and small neighborhood make fast convergence and exploration during genetic algorithm operators. The performance of the CGAMV ^{P} algorithm by the proposed model is examined on a numerical experiment and it shows that the designed model has successfully optimized the location of facilities and network flows.The efficiecy of the proposed model validate to deal with medical waste in hospitals and it could be applied on other manufacture. The result indicates that, the proposed model is applicable.

Keywords: *cellular genetic algorithm, hospital waste, mixed integer programming.*

1. Introduction

The concept of Closed Loop Supply Chain Reverse Logistics (CLSCRL) network, which means the combination of forward and reverse logistics, is now widely taking attention as a result of the recognition that both logistics have strong influence on supply chain performance and it is needed to manage these logistics simultaneously [14]. CLSCRL problems have been important issues for both researchers and practitioners from the last decade because of the customer expectations, increasing business competition, and regulatory pressures [8, 15, 16]. Comparing with reverse logistics, hospitals waste closed loop supply chain involves not only the reverse flows of the waste from the produced hospital to the manufacturers or related facilities such as gathering stations, but also the forward flows of waste from manufacturers to received hospitals [20, 5]. Numerous studies have been conducted on supply chain forward logistics [9, 10, 13, 7]. [18] have developed a mixed integer linear programming model for minimizing total costs of reverse logistics including transportation cost, fixed cost of operating, costs of collecting and processing centers, and operation costs of facilities to deal with

medical waste returned from hospitals to a manufacturer. In this paper, reverse logistics is indicated to as the procedure of logistics management involved in planning, managing, and controlling the influx of hospital waste for reuse or disposal of waste. The forward flow had been considered jointly with the reverse flow in their model, allowing the simultaneous definition of the optimal distribution and recovery networks.

A Mixed Variable Optimization (MVO) model is proposed for minimizing costs of hospital waste CLSCRL networks. The total costs for CLSCRL contain stable cost of opening the gathering stations, treating stations, distribution stations and received hospitals, shipped cost and processing cost. Finally a schedule is proposed of flows of hospital waste in the network. Four types of waste were presented in this paper (glass waste, metal waste, plastic waste and paper waste), but [18] used two types of waste (sharp waste and tissues waste), we did not use these waste as, when we asked in the Assiut hospital they replied to us that they burned sharp and tissues waste fearing of infection. A hybrid Cellular Genetic Algorithm (CGA) is used for solving the MVO model to be (CGAMV). Pattern Search Method (PSM) is added to the proposed method, to be ($CGAMV^P$) to make more intensification on the best solutions. The performance of the $CGAMV^P$ algorithm by the proposed model is examined on a numerical experiment and it shows that the designed model has successfully optimized the location of facilities and network flows. The result indicates that, the proposed model is applicable.

Cellular genetic algorithm (CGA) is a type of (GA) which is an efficient random search algorithm and it is a class of the applicable metaheuristics. The major advantage of metaheuristics is it offers a balance for finding a good solution (the global optimum) in a mild run time. GA works on a set of population, applying some stochastic operators (called genetic operators, e.g., selection, crossover (recombination), and mutation) [12]. CGA is a GA with dispersed population in which exchanges between individuals are restricted to closely neighbors. The main idea of this method is to permit the population of a specific structure clarified as a connected diagram, in which each individual interacts with his nearest neighbors. Individuals are mapped in a toroidal mesh, using this topology does not limit the region of the achieved solutions . In it, the boundary solutions of the grid are connected to the individuals located on the opposite borders in the identical row/column. The attained effect is a toroidal grid, so that all the individuals have precisely the same number of neighbors. There are six common neighborhood structures for every individual called L5, L9, C9, C13, C21 and C25 presented in [1, 4]. In CGA, cells can be renewed either asynchronously or synchronously [3]. There exist four types for sequentially updating the cells of a CGA: fixed line sweep (FLS), fixed random sweep (FRS), new random sweep (NRS), and uniform choice (UC) [1].

In the current work, the grid structure of $CGAMV^P$ method helps to get fast convergence. The selected chromosome with its small neighborhood make

exploration through genetic algorithm operators. The considered neighborhood in CGAMV^{P} method interest in making more intensification for every chromosome in population and also it tries to make some diversification to cover all parts in search space. CGAMV ^{P} method tries to solve the proposed model and propose a schedule of flows of hospital waste CLSCRL network for the opening gathering stations, treating stations, distribution stations and received hospitals.

The rest of present paper is arranged as follows: Section 2, defines the problem. Section 3 shows the model structure. Section 4 proposes a mixed variable optimization model of hospital waste closed loop supply chain reverse logistic network. Section 5 describes the CGAMV ^{P} method with their components in details for solving the proposed model. A numerical experiment is offered in Section 6. The computational results of CGAMV ^{P} method and scheduling of flows of the network are showed in Section 7. Finally, Section 8 which concludes the paper.

2. Problem Definition

In this study, The position problem of gathering stations, treating stations, distribution stations and received hospitals is illustrated, including stable cost, shipped cost and processing cost. In this network, there are G Gathering stations, T Treating stations, D Distribution stations and H Hospitals. Hospital waste is shipped from hospitals to gathering stations. Through some separating and collection in the gathering stations, the waste is again shipped from gathering stations to treating stations, where it will be completely antiseptic, disjointed, remanufactured or disposed. The hospital items were transported to the manufactory. The hospital waste was shipped to distribution stations. Finally, the reused hospital waste was turned to hospitals. Four types of waste is presented (glass waste, metal waste, plastic waste and paper waste), which these waste were treated and returned to received hospital. Regarding the nation wants these waste treating facilities as few as possible. The hospital waste CLSCRL network design containing five classes: hospitals, gathering stations, treating stations, manufactory and distribution stations.

The hospital waste CLSCRL network problem can be discussed as: offered the potential position and the capacities of the gathering stations, treating stations, distribution stations and received hospitals, the whole quantities of every type of the hospital waste produced by any hospital, the cost bodies for stable, shipping and other processing costs attached to the hospital waste, the model design must find out which possible gathering stations, treating stations, distribution stations and received hospitals, must be opened and how the hospital waste transferred so that CLSCRL network can obtain the minimum whole costs. Finally, a schedule of flows of hospital waste CLSCRL network for the opening gathering stations, treating stations distribution stations and received hospitals is proposed.

2.1. Model Structure

The underlying notations for model structure are the follows:

2.1.1 Control Variables

Control Variables Control Variables were used:

X_{hgk} *:* quantities transported from hospital h to gathering station g for waste k;

Y_{gtk} *:* quantities transported from gathering station g to treating station t for waste k ;

F_{tk} : quantities transported from treating station t to the manufactory for waste k;

S_{dk}*: quantities transported from manufactory to the distribution station d for waste k ;*

W_{dhk} *: returned quantities transported from distribution station d to the received hospital h for waste k;*

$$J_g = \begin{cases} 1; \text{ , if gathering station } g \text{ is open;} \\ 0; \text{ otherwise;} \end{cases}$$

$$R_t = \begin{cases} 1; \text{ , if treating station } t \text{ is open;} \\ 0; \text{ otherwise;} \end{cases}$$

$$\Lambda_d = \begin{cases} 1; \text{ , if distribution station } d \text{ is open;} \\ 0; \text{ otherwise;} \end{cases}$$

$$\Pi_h = \begin{cases} 1; \text{ , if hospital } h \text{ is open;} \\ 0; \text{ otherwise;} \end{cases}$$

2.1.2 Parameters

Parameters were used:
$H = \{1, 2, ..., H\}$, series of hospitals;
$G = \{1, 2, ..., G\}$, series of gathering stations;
$T = \{1, 2, ..., T\}$, series of treating stations;
$D = \{1, 2, ..., T\}$, series of distribution stations;
$K = \{1, 2, ..., K\}$, series of hospital waste kinds;
G_{max} : the maximum number of possible gathering stations;
T_{max} : the maximum number of possible treating stations;
D_{max} : the maximum number of possible distribution stations;
H_{max} : the maximum number of possible hospitals;
K : the number of the medical waste kinds;
Q_{hk} : the quantity from hospital *h* of hospital waste *k* ;
G_g^s : the stable costs of structure or capacity expansion of gathering station *g*;
T_t^s : the stable costs of structure or capacity expansion of treating station *t*;
Fs : the stable costs of capacity expansion of the manufactory;
D_d^s : the stable costs of structure or capacity expansion of distribution station *d*;
H_h^s : the stable costs of capacity expansion of the hospital *h*;
G_{gk}^p : the unit processing cost in gathering station *g* of hospital waste *k* ;
T_{tk}^p : the unit processing cost in treating station *t* of hospital waste *k* ;
F_k^p : the unit processing cost of hospital waste *k* in the manufactory;
D_{dk}^p : the unit processing cost in distribution station *d* of hospital waste *k* ;
H_{hk}^p: the unit processing cost in hospital *h* of hospital waste *k* ;
G_{gk}^i: the unit shipped cost of hospital waste *k* from hospital *h* to gathering station *g*;
T_{tk}^i : the unit shipped cost of hospital waste *k* from gathering station *g* to treating station *t*;
F_{tk}^i : the unit shipped cost of hospital waste *k* from treating station *t* to the manufactory;
D_{dk}^i : the unit shipped cost of hospital waste *k* from manufactory *g* to distribution station *d*;
H_{hk}^i : the unit shipped cost of hospital waste *k* from distribution station *d* to hospital *h*;
G_{gk}^c : the maximum capacity of gathering station *g* for hospital waste *k*;
T_{tk}^c : the maximum capacity of treating station *t* for hospital waste *k*;

D_d^c : the maximum capacity of distribution station *d* for hospital waste *k*;
H_h^c : the maximum capacity of hospital *h* for hospital waste *k*;
G_g^c : the maximum capacity of gathering station *g*;
T_t^c : the maximum capacity of treating station *t*;
D_d^c : the maximum capacity of distribution station *d*;
H_h^c : the maximum capacity of hospital *h*;
$\succ_k$: the discarding rate of hospital waste *k* in gathering stations;
η_k : the discarding rate of hospital waste *k* in treating stations;
α_k : the discarding rate of hospital waste *k* in hospitals;

3. Mathematical Model

We try to put mixed variable Programming for CLSCRL problem. Our mathematical model is made underlying simplification and assumption:
(1) Each Hospital could send medical waste to more than a gathering station. (2) the medical wastes do not include sharp waste. (3) the amount of medical waste in each hospital are independent. (4) The cost of any operation like transportation is fixed. (5) the capacity of gathering stations, treating stations and factories is limited.

Minimize $\boldsymbol{f(X;Y;F;S;W;J;R;\Pi;\Lambda)}$ =

$$\sum_{g=1}^{G} \boldsymbol{G_g^s J_g} + \sum_{t=1}^{T} \boldsymbol{T_t^s R_t} + \boldsymbol{F^s} + \sum_{d=1}^{D} \boldsymbol{D_d^s \Lambda_d} + \sum_{h=1}^{H} \boldsymbol{H_h^s \Pi_h} + \sum_{h=1}^{H}\sum_{g=1}^{G}\sum_{k=1}^{K} \left(\boldsymbol{G_{gk}^i} + \boldsymbol{G_{gk}^p} + (\mathbf{1} - \succ_k)\right) \boldsymbol{X_{hgk}} + \sum_{g=1}^{G}\sum_{t=1}^{T}\sum_{k=1}^{K} \left(\boldsymbol{T_{tk}^i} + \boldsymbol{T_{tk}^p} + (\mathbf{1} - \boldsymbol{\eta_k})\right) \boldsymbol{Y_{gtk}} + \sum_{t=1}^{T}\sum_{k=1}^{K} \left(\boldsymbol{F_{tk}^i} + \boldsymbol{F_{tk}^p}\right) \boldsymbol{F_{tk}} + \sum_{d=1}^{D}\sum_{k=1}^{K} \left(\boldsymbol{D_{dk}^i} + \boldsymbol{D_{dk}^p}\right) \boldsymbol{S_{dk}} + \sum_{d=1}^{D}\sum_{h=1}^{H}\sum_{k=1}^{K} \left(\boldsymbol{H_{hk}^i} + \boldsymbol{H_{hk}^p}(\mathbf{1} - \boldsymbol{\alpha_k})\right) \boldsymbol{W_{dhk}} \quad (1)$$

The objective function 1 minimizes whole CLSCRL costs consisted of stable cost in the first five terms in the objective function, shipped and processing costs in the rest terms of the objective function for opening gathering stations, treating stations, distribution stations and hospitals, respectively.

subject to:

$$\sum_{g=1}^{G} X_{hgk} = Q_{hk} \qquad (h = 1; \ldots..; H;\ k = 1; \ldots..; K) \quad (2)$$

Constraints (2) involves the equilibrium of hospital waste in produced hospital (the quantities transported from hospital to gathering stations equal the quantities produced from hospitals).

$$(1 - \succ_k) \sum_{h=1}^{H} X_{hgk} = \sum_{t=1}^{T} Y_{gtk} \quad (g = 1; \ldots; G;\ k = 1; \ldots; K) \quad (3)$$

Constraints (3) involves the equilibrium of hospital waste in gathering stations (the quantities transported from hospital to

gathering stations product of the discarding rate of hospital waste in gathering stations equal the quantities transported from gathering stations to treating stations).

$$(1-\eta_k)\sum_{g=1}^{G}\sum_{t=1}^{T}Y_{gtk}=\sum_{t=1}^{T}F_{tk}\quad(k=1,\dots\dots,K)\quad(4)$$

Constraints (4) involves the equilibrium of hospital waste in treating stations (the quantities transported from gathering stations to treating stations product of the discarding rate of hospital waste in treating stations equal the quantities transported from treating stations to manufactory).

$$\sum_{t=1}^{T}F_{tk}=\sum_{d=1}^{D}S_{dk}\qquad(k=1\dots\dots,k)\qquad(5)$$

Constraints (5) involves the equilibrium of hospital waste in manufactory (the quantities transported from treating stations to manufactory equal the quantities transported from manufactory to distribution stations).

$$S_{dk}=\sum_{h=1}^{H}W_{dhk}\qquad(d=1,\dots\dots..D\,;k=1\dots\dots,K)$$

Constraints (6) involves the equilibrium of hospital waste in distribution stations (the quantities transported from manufactory to distribution stations equal the returned quantities transported from distribution stations to received hospital).

$$(1-\alpha_k)\sum_{d=1}^{D}\sum_{h=1}^{H}W_{dhk}=Q_{hk}\quad(k=1,\dots,K)\qquad(7)$$

Constraints (7) involves the equilibrium of hospital waste in received hospitals (the quantities transported from distribution stations to received hospital product of the discarding rate of hospital waste in hospitals equal the quantities produced from hospitals).

$$\sum_{h=1}^{H}\sum_{k=1}^{K}X_{hgk}\le G_g^cJ_g\quad(g=1\dots,G)\qquad(8)$$

Constraint (8) includes that the whole amount of waste transported to each gathering station cannot skipped the capacity of the gathering station *g* keeping them.

$$\sum_{g=1}^{G}\sum_{k=1}^{K}Y_{gtk}\le T_t^cR_t\quad(t=1\dots,T)\qquad(9)$$

Constraint (9) includes that the whole amount of waste transported to each treating station cannot skipped the capacity of the treating station *g* keeping them.

$$\sum_{k=1}^{K}S_{dk}\le D_d^cR_t\quad(d=1\dots,D)\qquad(10)$$

Constraint (10) includes that the whole amount of waste transported to each distribution station cannot skipped the capacity of the distribution station *g* keeping them.

$$\sum_{d=1}^{D}\sum_{k=1}^{K}W_{dhk}\le H_h^c\Pi_h\quad(h=1\dots,H)\qquad(11)$$

Constraint (11) includes that the whole returned amount of waste transported to hospital cannot skipped the capacity of the hospital *h* keeping them.

$$\sum_{h=1}^{H}X_{hgk}\le G_{gk}^cJ_g\quad(g=1\dots,G\,;k=1,\dots K)\qquad(12)$$

Constraint (12) includes that the quantity of waste *k* transported to each gathering station cannot skipped the capacity of the station keeping it.

$$\sum_{g=1}^{G}Y_{gtk}\le T_{tk}^cR_t\quad(t=1\dots,T\,;k=1,\dots K)\qquad(13)$$

Constraint (13) includes that the quantity of waste *k* transported to each treating station cannot skipped the capacity of the station keeping it.

$$S_{dk} \leq D_{dk}^{c}\Lambda_d \quad (d = 1 \dots, D\,; k = 1, \dots K) \qquad (14)$$

Constraint (14) includes that the quantity of waste *k* transported to each distribution station cannot skipped the capacity of the station keeping it.

$$\sum_{d=1}^{D} W_{hk} \leq H_{hk}^{c}\Pi_h \quad (h = 1 \dots, H\,; k = 1, \dots K) \qquad (15)$$

Constraint (15) includes that the returned quantity of waste *k* transported to each hospital cannot skipped the capacity of the hospital keeping it.

$$\sum_{g=1}^{G} J_g \leq G_{max} \qquad (16)$$

Constraint (16) the number of the opening gathering stations cannot exceed the maximum number of possible gathering stations.

$$\sum_{t=1}^{T} R_t \leq T_{max} \qquad (17)$$

Constraint (17) the number of the opening treating stations cannot exceed the maximum number of possible treating stations.

$$\sum_{d=1}^{D} \Lambda_d \leq D_{max} \qquad (18)$$

Constraint (18) the number of the opening distribution stations cannot exceed the maximum number of possible distribution stations.

$$\sum_{h=1}^{H} \Pi_h \leq H_{max} \qquad (19)$$

Constraint (19) the number of the opening hospitals cannot exceed the maximum number of possible hospitals)

$$X_{hgk}, Y_{gtk}, F_{tk}, S_{dk}, W_{dhk} \geq 0 \qquad (20)$$

Constraint (20) confirms that the non-negativity of control variables X_{hgk} , Y_{gtk} , F_{gk} , S_{dk} and W_{dhk} .

$$J_g \in \{0,1\} \qquad (21)$$

Constraint (21) confirms that the binary of control variables J_g , whether the gathering station is opened (=1) or closed (=0) .

$$R_t \in \{0,1\} \qquad (22)$$

Constraint (22) confirms that the binary of control variables *Rt*, whether the treating station is opened (=1) or closed (=0) .

$$\Lambda_d \in \{0,1\} \qquad (23)$$

Constraint (23) confirms that the binary of control variables Λ_d, whether the distribution station is opened (=1) or closed (=0).

$$\Pi_h \in \{0,1\} \qquad (24)$$

Constraint (24) confirms that the binary of control variables Π_h, represent whether the received hospital is opened (=1) or closed (=0))

4. Solution Approach: Cellular Genetic Algorithm

Cellular Genetic Algorithm for Mixed Variable Optimization (CGAMV) is implemented as randomly generated individuals to map on a 2-D toroidal grid. Focusing on a type of asynchronous FLS cellular genetic algorithm to update the cells. The selected individual and its neighbors defined as small population P, in which the genetic factors (parent selection, recombination and mutation) are applied that make exploration in every selected individual. The selected individual is updated by the best chromosome in P after GA operators. Therefore, the diversification

process is added to cover the most space in the search region. Finally, a pattern search is added to CGAMV method to be CGAMV P to get more intensification on the best solutions. The proposed method is illustrated as follows:

4.1. Population Encoding

Chromosomes (Individuals) in initial population are created by uniform distributed inside the search region bounded by [l, u], representing a good solutions. Obviously, X_{hgk} , Y_{gtk} , F_{gk} , S_{dk} and W_{dhk} are continuous value variables while Jg , R_t, Λ_d and Π_h are binary value variables. Each chromosome initialized is based on one dimensional vector which contains continuous variables, representing quantity transported from hospital h to gathering station g, from gathering station g to treating station t, from treating station t to the manufactory, from manufactory to distribution station d and from distribution station d to hospital t for waste k and binary variables, related to gathering stations, treating stations, distribution stations and received hospitals. The chromosome contains six hospitals, five gathering stations, three treating stations, one manufactory and three distribution stations. Thus, each chromosome is represented as a (6*5+5*3+3*1+1*3+3*6+5+3+3+6) vector, where (6*5) genes represent the hospital waste from six hospitals to five gathering stations; (5*3) genes represent the hospital waste from five gathering stations to three treating stations; (3*1) genes represent the hospital waste from three treating stations to manufactory; (1*3) genes represent the hospital waste from manufactory to three distribution stations; (3*6) genes represent the hospital waste from three distribution stations to six hospitals; (5) genes represent the possible gathering station; (3) genes represent the possible treating station; (3) genes represent the possible distribution station; (6) genes represent the possible received hospital. See the representation of a chromosome in Figure 1.

4.2. Evaluation Fitness Function

This evaluation procedure is used for comparing one solution with others in the population. In this paper, penalty method [11] is used to deal with constraint.

Figure 1: A representation chromosome scheme problem.

4.3. Mapping

Individuals are mapped on 2-D toroidal grid from best to worst. In the case of a population of size μ arranged in a toroidal grid of size $\sqrt{\mu} \times \sqrt{\mu}$ (assuming μ odd) [6]. An chromosome (individual) represented as **z** = (X_{111}, ..., X_{654}, Y_{111}, ..., Y_{534}, F_{11}, ..., F_{34}, S_{11}, ..., S_{34}, W_{111}, ..., W_{364}, G_1, ..., G_5, T_1, ..., T_3, D_1, ...D_3, H_1, ..., H_6) mapped on the grid as position (**z**) is the pair of coordinates (*i*, *j*) in the two-dimensional grid.

4.4. Neighborhood

In CGA, every individual select some of neighborhoods with various methods. In our method, the neighborhoods are selected by linear9 (L9) process. The neighborhood is the 8 nearest individuals composed by the center individual plus its two nearest

neighbors in horizontal and vertical directions as shown in Figure 2 [1]. The considered individual with its neighborhood became a small population P. The genetic factors are applied to this P.

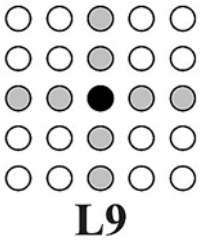


Figure 2: The L9 neighborhood

4.5. Parent Selection

The parent selection mechanism produces an intermediate population, say Ṕ from the current population P . Ṕ has the same size as P but an individual can be present in Ṕ more than once. Indeed, individuals in Ṕ are copies of individuals in P depending on their ranking: the higher ranking an individual has, the more the probability that it will be copied. This process is repeated until Ṕ is full while the chosen individual is already not removed. The linear ranking selection mechanism [2] is then applied to choose the members in P´.

4.6. Recombination

The recombination operation has an exploration tendency, and therefore it is not applied to all parents. First, for each individual in the intermediate population Ṕ, the crossover operation chooses a random number from the interval (0, 1). If the chosen number is less than the crossover probability $p_c \in (0, 1)$, the individual is added to the parent pool. After that, two parents **p**1 and **p**2 from the parent pool are randomly selected and mated to produce two children **c**1 and **c**2, which are then placed in the children pool. These procedures are repeated until all parents are mated. A recombined child is calculated by another random number $\lambda \in (0, 1)$. Let us note that a parent is selected only once, and if the total number of parents inside the parent pool is odd, then the unfortunate last parent added into the pool is not considered for the mating procedure. The following procedure describes the CGAMV recombination operation precisely based on One Point Crossover.

Procedure 1 *One Point Crossover*(**p**1, **p**2, **o**1, **o**2)

Let n_x the number of continuous variables, n_y the number of discrete variables and $n = n_x + n_y$.

1. Generate an integer random number $r \in (1, n)$.
2. Let $\mathbf{p}^1 = (\mathbf{x}^1, \mathbf{y}^1)$ and $\mathbf{p}^2 = (\mathbf{x}^2, \mathbf{y}^2)$.
3. Calculate the recombined offspring $\mathbf{O}^1 = (\mathbf{r}^1, \mathbf{s}^1)$ and $\mathbf{O}^2 = (\mathbf{r}^2, \mathbf{s}^2)$, where
   $\mathbf{O}^1(1 : n) = \mathbf{p}^1(1 : r) + \mathbf{p}^2(r + 1 : n)$,
   $\mathbf{O}^2(1 : n) = \mathbf{p}^2(1 : r) + \mathbf{p}^1(r + 1 : n)$.
4. return.

4.7. Mutation

For each gene in the intermediate population *Ṕ*. A random number from the interval (0, 1) is associated. If the associated number is less than the mutation probability $p_m \in (0, 1)$, then the individual will be mutated. The mutated child is computed through the following procedure. Let the numbers of the selected gene and chromosome are λ and θ, respectively.

Then, the mutated offspring **z** is computed among the following:

Procedure 2 *Uniform Mutation*

1. If the selected gene λ is continues, update $\mathbf{z}\theta$ by setting $\boldsymbol{x}_{\lambda}^{\boldsymbol{\theta}}$ = $\mathbf{l}_{\mathbf{x}\lambda} + \gamma(\mathbf{u}_{\mathbf{x}\lambda} - \mathbf{l}_{\mathbf{x}\lambda})$.
2. If the selected gene λ is integer, update $\mathbf{z}\theta$ by setting $\boldsymbol{y}_{\lambda}^{\boldsymbol{\theta}}$ = $\mathbf{l}_{\mathbf{y}\lambda} + \gamma(\mathbf{u}_{\mathbf{y}\lambda} - \mathbf{l}_{\mathbf{y}\lambda})$, where $\gamma \in (0, 1)$.

4.8. Replacement

Our current individual is updated by the best offspring in the small population P after the GA operators (selection, recombination and mutation process).

Procedure 3 *Replacement*

1. Sort the small population from best solution to worst one.
2. Update the individual with the best one in the population P.

4.9. Intensification

CGAMV uses a local search method to improve the best solution is found. Some elite off springs of the generation are improved by adding pattern search schemes. Specially, some reformed versions of the PSM are invoked to deal with the present model. The inputs and parameters of PSM are reset to contain both continuous and integer settings. Some modification is added on pattern search method to decrease the rate of cost function. Modified pattern search use the set D in distinct way.

$D = \{(\pm 1)h_{e1}, \dots, (\pm 1)h_{en}\}$,

where h is random number h ∈ {1, 2}, therefore, modified pattern search generate only n + 1 points at every iteration.

Procedure 4 *Generalized Pattern Search*

1. **Initialization.** Select an initial solution $\mathbf{x}_0$, select a positive spanning directions *D*, select a step size $\Delta_0 > 0$ and set the counter number k:= 0.
2. **Search Step.** Compute the grid $W_k = \{\mathbf{x} : \mathbf{x} = \mathbf{x}_k \pm \Delta_k \mathbf{d}\ \mathbf{z}, \mathbf{d} \in D, \mathbf{z} \in \boldsymbol{z}_{+}^{|D|}\}$, get an improved point from *Wk* . If an improvement is achieved go to Step 4.
3. **Poll Step.** Choose the search direction set $D_k \subset D$ to compute the poll set $P_k = \{\mathbf{x} : \mathbf{x} = \mathbf{x}k \pm \Delta_k \mathbf{d}, d \in D_k\}$,. Evaluate the cost function at all points in P_k .
4. **Update Step.** If an improved point obtained in Step 2 or 3, set $\mathbf{x}k+1$ equal to this improved point, and set $\Delta_{k+1} \geq \Delta_k$. Otherwise, set $\mathbf{x}_{k+1} := \mathbf{x}_k$, and $\Delta_{k+1} < \Delta_k$.
5. **Termination Conditions.** If the termination conditions are satisfied, then stop. Otherwise, set $k := k + 1$, and go to Step 2.

4.10. CGAMV ^{P} Algorithm

The previous components for the *CGAMV P* are shown in Algorithm 1 and in Fig 3.

Algorithm 1: *CGAMV P* Algorithm

1. **Initialization.** Set the crossover and mutation probabilities $p_c \in (0, 1)$ and $p_m \in (0, 1)$, respectively. Set the generation counter $t := 0$.
2. **Population Encoding.** Generate an initial population P_0.
3. **Evaluation.** Evaluate the fitness function f and constraint violation function for all individuals in P_t.
4. **Mapping.** Map all the individuals in P_t on two dimensional grid.
5. **Neighborhood.** For every individuals in the grid calculate neighborhood in small population *P* .

6. **Parent Selection.** Select an intermediate population $\acute{P}$ from the current population P using the linear ranking selection mechanism.
7. **Recombination.** Associate a random number from (0, 1) with each individual in $\acute{P}$ and add this individual to the parent pool PP if the associated number is less than p_c. Repeat the following Steps 7.1 and 7.2 until all parents in PP are mated:

   7.1 Choose two parents **p**1 and **p**2 from PP . Mate **p**1 and **p**2 to reproduce children **c**1 and **c**2.

   7.2 Update the children pool CP by $CP := CP \cup \{\mathbf{c}1; \mathbf{c}2\}$ and update PP by $PP := PP/\{\mathbf{p}1; \mathbf{p}2\}$.
8. **Mutation.** Associate a random number from (0, 1) with each gene for each individual in $\acute{P}$. Mutate the individuals which their associated number less than p_m and add the mutated individual to CP .
9. **Replacement.** Replace position(individual) by best child.
10. **Intensification.** Apply a local search method to improve the best solution is found.
11. **Stopping Condition.** If stopping conditions are satisfied, then terminate. Otherwise, set $t := t + 1$, and go to Step 5.

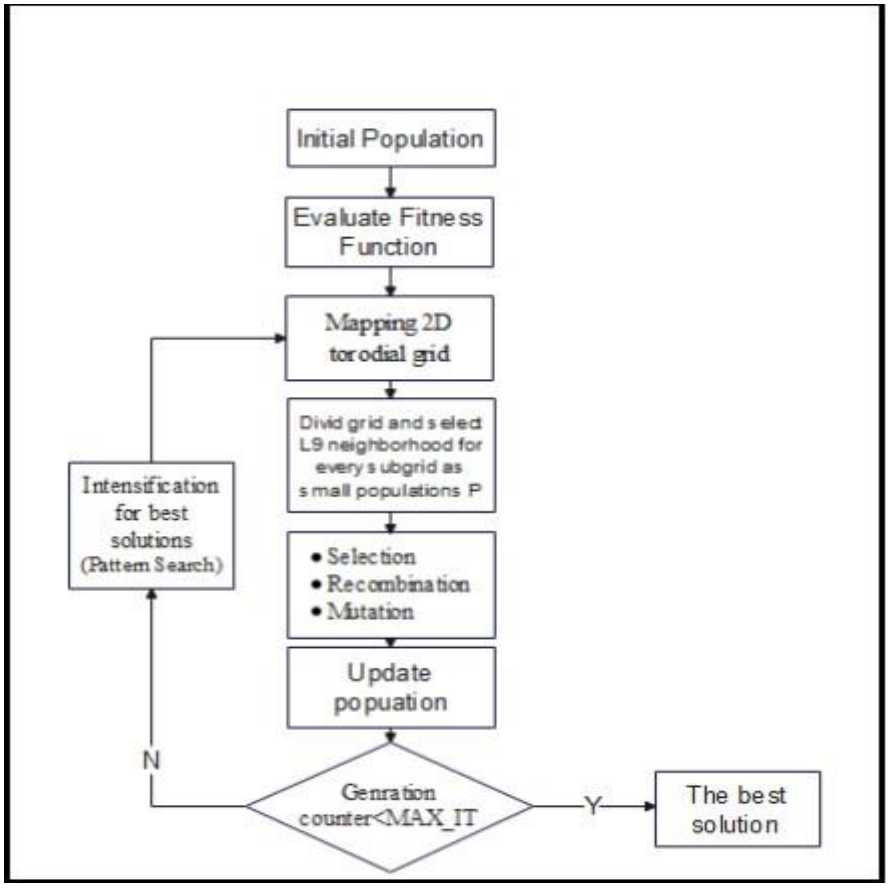


Fig 3: CGAMV ^{P} Algorithm

5. Experiment and Discussion

The CGAMV ^{P} method is applied to solve our mathematical model to demonstrate and verify the model importance

For describing the model, the model is applied using a numerical experiment. In this experiment five gathering stations G1, G2, G3, G4, G5, three treating stations T 1, T 2, T 3, one manufactory, three distribution station D1, D2, D3 and six hospitals H1, H2, H3, H4, H5, H6 are used. Assuming the numerical test in [18] and we used the rest of numbers stochastic, because these numbers are difficult to get in Egypt.

- Four classes of hospital waste I, II, III and IV (glass waste, metal waste, plastic waste and paper waste) created from each hospital see Table 1.
- Table 2 shows the stable cost of manufactory, gathering stations, treating stations, distribution stations and received hospital.
- The processing cost and capacity of gathering stations are exhibited in Table 3.
- The processing cost and capacity of treating stations are displayed in Table 4.
- The processing cost and capacity of distribution stations are displayed in Table 5.
- The processing cost and capacity of received hospitals are displayed in Table 6.
- The processing cost of manufactory shows in Table 8.
- The maximum capacity of gathering stations, treating stations, distribution stations and received hospitals are exhibited in Table 7.
- The discarding rates of hospital waste in gathering stations, treating stations and received hospitals are the same see Table 9.
- The shipped cost attached to the quantity of hospital waste is shown Table 10-12.

Table 1: Quantity of hospital waste from hospitals T(Ton)

Hospitals	WasteI	WasteII	WasteIII	WasteIV
1	45	20	30	40
2	70	40	60	50
3	95	75	80	90
4	60	45	50	55
5	80	55	60	75
6	55	20	25	50

Table 2: Stable cost (Pound/Ton)

Manufactory	1	50000
Gathering stations	1	8500
	2	10000
	3	15000
	4	12000
	5	10000
Treating stations	1	4000
	2	45000
	3	50000
Distribution stations	1	30000
	2	40000
	3	50000
Hospitals	1	40000
	2	45000
	3	50000
	4	60000
	5	70000
	6	80000

Table 3: Processing Cost and capacity of gathering stations(Pound/Ton)

Processing Cost				
Gathering stations	WasteI	WasteII	WasteIII	WasteIV
1	7	3	2	6
2	5	3	2	4
3	5	1.5	1	4.5
4	6	2.5	1.5	5
5	4	3	2	3
Capacity				
1	100	80	70	90
2	150	120	110	140
3	150	150	140	140

Table 4: Processing Cost and capacity of treating stations(Pound/Ton)

Processing Cost				
Treating	WasteI	WasteII	WasteIII	WasteIV
1	9	7	8	6
2	7	5	6	4
3	5	4	4	3
Capacity				
1	200	120	180	160
2	250	170	190	200
3	300	220	250	270

Table 6: Processing Cost and capacity of hospitals(Pound/Ton)

4	130	150	120	140
5	120	100	110	90

Table 5: Processing Cost and capacity of distribution stations(Pound/Ton)

Processing Cost				
Distribution stations	WasteI	WasteII	WasteIII	WasteIV
1	8	7	6	3
2	6	5	4	2
3	7	4	3	2
Capacity				
1	200	130	100	200
2	240	150	120	320
3	290	200	220	300

Processing Cost				
Hospitals	WasteI	WasteII	WasteIII	WasteIV
1	9	7	8	6
2	7	5	3	2
3	5	4	3	2
4	4	2.5	1.5	5
5	5	3	4	6
6	8	9	7	5
Capacity				
1	200	120	150	170
2	250	170	160	130
3	300	220	200	100
4	350	150	110	100
5	400	200	300	100
6	450	240	320	120

Table 7: Maximum capacity (Pound/Ton)

Gathering stations	1	160
	2	260
	3	280
	4	260
	5	200
Treating stations	1	300
	2	400
	3	500
Distribution stations	1	150
	2	250
	3	350
Hospitals	1	300
	2	400

Table 8: Processing Cost of manufactory(Pound/Ton) in first and second rows, discarding rates in gathering stations in third and forth rows and finally, shipped cost from treating stations to the manufactory (Pound) in the residue of the table

Manufactory	WasteI	WasteII	WasteIII	WasteIV
1	8	6	7	9
Discarding rates	WasteI	WasteII	WasteIII	WasteIV
λ, η, α	0.8	0.7	0.6	0.9
Treating stations	WasteI	WasteII	WasteIII	WasteIV

	3	500
	4	600
	5	700
	6	800

1	6	1	2	5
2	3	5	2	4
3	5	7	4	6

Table 9: Shipped cost from hospitals to gathering stations(Pound)

Hospitals	Waste	Gathering stations				
		1	2	3	4	5
1	I	5	10	8	7	6.5
	II	7	9	10	10	13
	III	4	9	7	6	5.5
	IV	6	8	9	9	12
2	I	3	5	4	5	10
	II	8	8.5	3	6	6.5
	III	2	4	3	4	9
	IV	7	7.5	2	5	5.5
3	I	6	4	5	3	9
	II	10	8.5	6	5	7
	III	5	3	4	2	8
	IV	9	7.5	5	4	6
4	I	7	1	10	12	6
	II	12	11	8	9	10
	III	6	2	9	11	5
	IV	11	10	7	8	9
5	I	4	3	6	9	5
	II	5.5	8	7	10	8
	III	3	2	5	8	4
	IV	4.5	7	6	9	7
6	I	3	9	7	6	10
	II	8	12	8	9	10
	III	2	8	6	5	9
	IV	7	11	7	8	9

Table 10: Shipped cost from gathering stations to treating stations(Pound)

Treating stations	Waste	Gathering stations		
		1	2	3
1	I	4	5	2
	II	6	7	5
	III	1	2	3
	IV	4	5	6
2	I	6.5	5	7
	II	7	8	5
	III	7	8	9
	IV	1	2	3
3	I	6.5	4	5.5
	II	8	6.5	5
	III	4	5	6
	IV	7	8	9
4	I	7.5	3	8
	II	10	8	8
	III	1	2	3
	IV	4	5	6
5	I	5	4	4
	II	5	7	6
	III	7	8	9
	IV	1	2	3

Table 11: Shipped cost from distribution stations to hospitals(Pound)

Distribution stations	Waste	Hospitals					
		1	2	3	4	5	6
1	I	6	1	2	5	4	5
	II	5	7	8	3	6	7
	III	3	5	4	2	3	1
	IV	4	6	7	6	1	2
2	I	7	9	8	5	9	10
	II	3	5	2	1	6	7
	III	6	1	2	5	4	5
	IV	5	7	8	3	6	7
3	I	3	5	4	2	3	1
	II	4	6	7	6	1	2
	III	7	9	8	5	9	10
	IV	3	5	2	1	6	7

Table 12: Shipped cost from manufactory to distribution stations(Pound)

Distribution stations	WasteI	WasteII	WasteIII	WasteIV
1	8	7	6	3
2	6	5	4	2
3	7	4	3	2

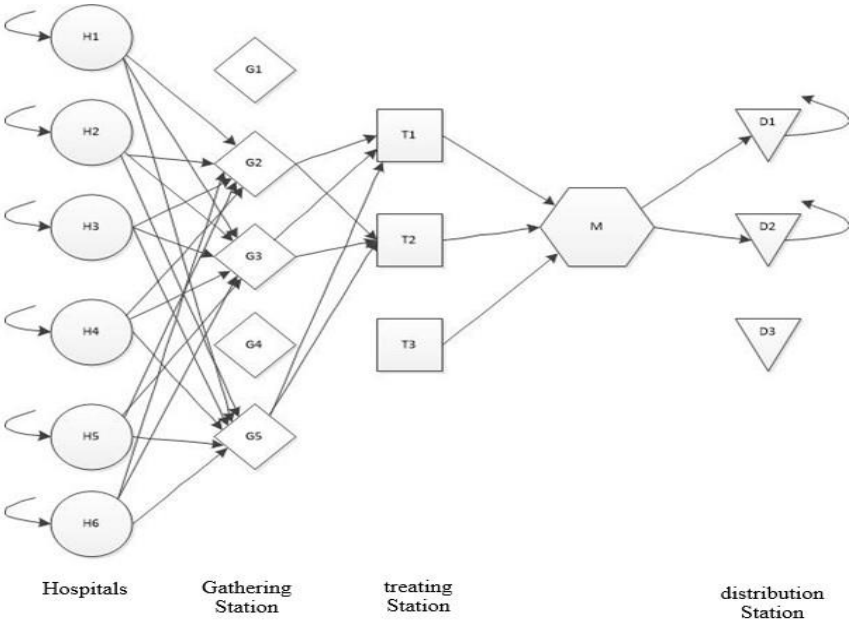


Fig 4: A schedule of hospital waste closed loop supply chain reverse logistics network

5.1. Parameters Setting

The CGAMV ^{P} method is programmed in MATLAB. In this Section, a discussion of the parameters setting and the efficiency analysis of the CGAMV ^{P} algorithm as the following descriptions:

The $CGAMV^{P}$ parameters are setting as Table 17 which discussed to cover the description of $CGAMV^{P}$ algorithm stated in the previous section. Assume P_{size} is the population size, N_i is the digit of pattern search iterations and N_g is the number of generations.

- The population size is set to be 361 population, since it is almost the smallest value which can give good results within an acceptable number of function evolutions.
- p_c = 0.8, since is almost the standard setting of the crossover probability
- p_m = 0.01, since is almost the standard setting of the mutation probability.
- The number N_L of local search iterations is set equal to 20.
- The ranking parameter is set to be 5.
- $CGAMV^{P}$ is forced to be terminated if the number of generations exceeds 200.

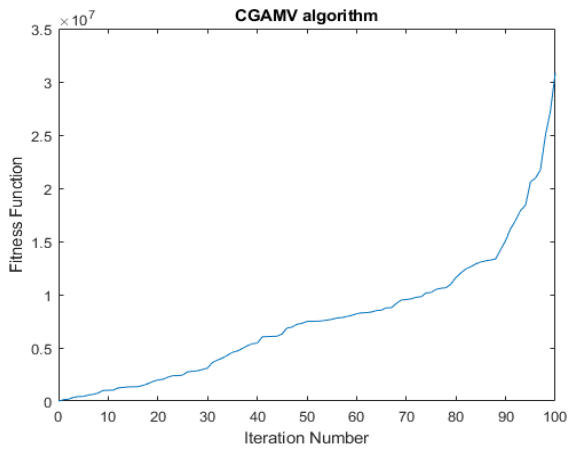
P_{size}	19 × 19 (361 Individuals)
Neighborhood	L9
Selection	Linear Ranking
Crossover	one Point
Cross. Prob	0.8
Mutation	Uniform
Mutation. Prob	0.01
N_i for every best solution	20
Termination Condition	N_g *exceeds 200*

Table 13: Parameters used for the $CGAMV^{P}$

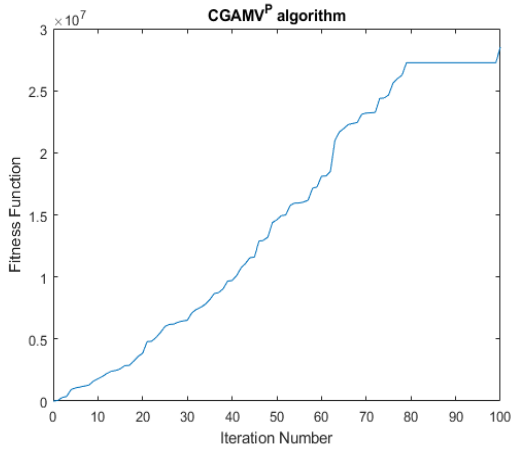
5.2 Efficiency Analysis

In this section, the performance of *CGAMV P* method is discussed on the numerical experiment, which are shown in the previous section. *CGAMV P* method is proposed to the CLSCRL model which is considered as NP-hard problems. The numerical results are discussed through 1 independent run. Making the value of our model as real problem so, the results are taken when using high limitation, then we get the fitness function value as 7.2606e10 and a schedule of the potential gathering stations 2, 3, 5, treating stations 1, 2, distribution stations 1, 2, received hospitals 1, 2, 3, 4, 5, 6 are elected for the flows of the hospital waste network is obtained clarified in Figure (4).

Figure (5) show the performance of CGAMV algorithm without pattern search as an intensification procedure. That performance demonstrates our algorithm without pattern search method takes more iterations to reach the best solution. The performance of CGAMV P is shown in Figure (6) and that clarifies the importance of pattern search to find the best solution faster. The bigger value of the fitness function is neglected due to two main reasons: the algorithm solves the mathematical method which is considered as NP-hard problem and the huge number of parameters in our model. We can put the parameters output values as a graph such as it presented in Figure (4) which will make it easier for ordinary users.



Fig(5) CGAMV algorithm



Fig(5) CGAMV ^{P} algorithm

6. Conclusions and Future Work

In this research, the hospital waste closed loop supply chain reverse logistics network structure is discussed, created from the hospitals to manufactories. A mixed variable optimization model is created for minimizing the total costs include stable cost of opening the gathering stations, treating stations, distribution stations and received hospitals, shipped cost and processing cost. A hybrid cellular genetic algorithm with pattern search has been used to get a good solution with the proposed model. A scheduling of flows of the network is proposed of the opening gathering

stations, treating stations, distribution stations and received hospitals. The selection procedure in the grid of CGAMV ^{P} method make more intensification on every chromosome and CGAMV ^{P} tries to cover the search space by diversification. The computational results for the numerical experiment are shown promising of the proposed method to deal with NP hard problem. In future research, we maybe discuss other metaheuristics methods to deal with our mathematical model. Also, we will try to apply the mathematical model in various places as universities and factories

References

1. Dahi, Zakaria Abdelmoiz, and Enrique Alba. "The grid-to-neighbourhood relationship in cellular GAs: from design to solving complex problems." Soft Computing 24.5 (2020): 3569-3589.
2. Alba, E., and Dorronsoro, B. Cellular genetic algorithms, vol. 42. Springer Science & Business Media, 2009
3. Chakraborty, M., and Chakraborty, U. K. Branching process analysis of linear ranking and binary tournament selection in genetic algorithms. Journal of computing and information technology 7, 2 (1999), 107–113.
4. Dorronsoro, B., Alba, E., Giacobini, M., and Tomassini, M. The influence of grid shape and asynchronicity on cellular evolutionary algorithms. In Evolutionary Computation, 2004. CEC2004. Congress on (2004), vol. 2, IEEE, pp. 2152–2158.
5. Dorronsoro, B., and Bouvry, P. Adaptive neighborhoods for cellular genetic algorithms. In Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW), 2011 IEEE International Symposium on (2011), IEEE, pp. 388–394.
6. Easwaran, G., and Uster, H. ¨ A closed-loop supply chain network design problem with integrated forward and reverse channel decisions. Iie Transactions 42, 11 (2010), 779–792.
7. Giacobini, M., Alba, E., Tettamanzi, A., and Tomassini, M. Modeling selection intensity for toroidal cellular evolutionary algorithms. In 22 Genetic and Evolutionary Computation–GECCO 2004 (2004), Springer, pp. 1138–1149.
8. Govindan, K., Soleimani, H., and Kannan, D. Reverse logistics and closed-loop supply chain: A comprehensive review to explore the future. European Journal of Operational Research 240, 3 (2015), 603–626.
9. Ritola, Ilkka, Harold Krikke, and Marjolein CJ Caniëls. "Learning-based dynamic capabilities in closed-loop supply chains: an expert study." *The International Journal of Logistics Management* (2022).
10. Guide Jr, V. D. R., and Van Wassenhove, L. N. Or forum—the evolution of closed-loop supply chain research. Operations research 57, 1 (2009), 10–18.
11. Kannan, G., Sasikumar, P., and Devika, K. A genetic algorithm approach for solving a closed loop supply chain model: A case of battery recycling. Applied Mathematical Modelling 34, 3 (2010), 655–670.
12. Liu, J., Teo, K. L., Wang, X., and Wu, C. An exact penalty functionbased differential search algorithm for constrained global optimization. Soft Computing 20, 4 (2016), 1305–1313.
13. Michalewicz, Z. Genetic algorithms+ data structures evolution programs. Springer Science & Business Media, 2013.
14. Pishvaee, M. S., Rabbani, M., and Torabi, S. A. A robust optimization approach to closed-loop supply chain network design under uncertainty. Applied Mathematical Modelling 35, 2 (2011), 637–649.
15. Saha, A., Asadujjaman, M., and Asaduzzaman, M. A mixed integer linear programming model for solving closed loop supply chain problems. Journal of Modern Science and Technology 5, 1 (2017).
16. Savaskan, R. C., Bhattacharya, S., and Van Wassenhove, L. N. Closed-loop supply chain

models with product remanufacturing. Management science 50, 2 (2004), 239–252.

17. Schultmann, F., Zumkeller, M., and Rentz, O. Modeling reverse logistic tasks within closed-loop supply chains: An example from the automotive industry. European journal of operational research 171, 3 (2006), 1033–1050.
18. Shi, Lihong, et al. Network model and optimization of medical waste reverse logistics by improved genetic algorithm. *International Symposium on Intelligence Computation and Applications*. Springer, Berlin, Heidelberg, 2009.
19. Shadkam, Elham. "Cuckoo optimization algorithm in reverse logistics: A network design for COVID-19 waste management." *Waste Management & Research* (2021): 0734242X211003947.
20. Vlachos, D., Georgiadis, P., and Iakovou, E. A system dynamics model for dynamic capacity planning of remanufacturing in closed-loop supply chains. Computers & Operations Research 34, 2 (2007), 367–394