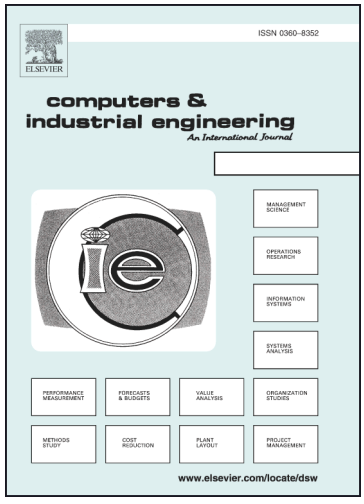
Accepted Manuscript



A Location- Routing Problem for Cross-docking Networks: A Biogeography-based Optimization Algorithm

Asefeh Hasani Goodarzi, Seyed Hessameddin Zegordi

PII: S0360-8352(16)30409-0
DOI: http://dx.doi.org/10.1016/j.cie.2016.10.023
Reference: CAIE 4508

To appear in: *Computers & Industrial Engineering*

Received Date: 6 January 2016
Revised Date: 25 October 2016
Accepted Date: 26 October 2016

Please cite this article as: Goodarzi, A.H., Zegordi, S.H., A Location- Routing Problem for Cross-docking Networks: A Biogeography-based Optimization Algorithm, *Computers & Industrial Engineering* (2016), doi: http://dx.doi.org/10.1016/j.cie.2016.10.023

A Location- Routing Problem for Cross-docking Networks: A Biogeography-based Optimization Algorithm

Asefeh Hasani Goodarzi[1], Seyed Hessameddin Zegordi[11]

[1] *Faculty of Industrial & Systems Engineering , Tarbiat Modares University, P.O.Box 14115- 143,Tehran, Iran.*

Asefeh.hasani@modares.ac.ir, Zegordi@modares.ac.ir

[1] Corresponding author. Tel: +98 21 82883394; Fax: +98 21 88005040.

A Location- Routing Problem for Cross-docking Networks: A Biogeography-based Optimization Algorithm

Abstract

This paper considers a location-routing problem in a distribution network with a set of part suppliers, cross-docking centers and assembly plants known as customers. We develop a mixed integer non-linear programming formulation for the problem in which the location for establishing the cross-docks is determined while simultaneously a fleet of vehicles are applied to transport goods from suppliers to the assembly plants via two transportation strategies: direct shipment and shipment through cross-dock (indirect shipment). In the second strategy, it is possible to have routes between suppliers. Not considering two problems of location and distribution planning simultaneously would result in increasing the costs of supplying parts since the transportation strategy has a huge effect on location of cross-docks. In the other words, if some loads can be directly shipped, then this kind of loads should not be taken into account in determining cross-docks location. Thus, a location- routing problem is presented for cross-docking system in this paper. The goal is to determine the location of cross-docks, allocating suppliers to them and routing decisions, so that the location cost and total shipping cost in the network are minimized, considering variable cost of servicing parts passed through cross-docks. The proposed model is NP-hard based on literature. Thus, a metaheuristic algorithm named Biogeography-based optimization (BBO) is utilized to solve the problem. In order to evaluate its efficiency, BBO results are compared with those of PSO, which is a well-known algorithm in the literature. Solving numerical examples for small size problem instances illustrates that the solving approach performs with a negligible gap relative to GAMS, while it performs much better than PSO in most cases in terms of total cost of the network and computational time.

Keywords: Cross-docking; Location-Routing Problem (LRP); Direct shipment; Distribution network; mixed integer non-linear programming.

1. Introduction

Nowadays distribution strategy is a fundamental element in each supply chain. As acclaimed by Apte and Viswanathan (2000) about 30% of goods prices are incurred in the distribution process. Thus, enhancing distribution strategies along with satisfying customer's demands is a vital issue. There are several different strategies in distribution networks which generally consist of direct shipping, milk runs, cross-docking and tailored networks (Chopra and Meindl, 2001). If a shipment is about to a full truckload (FTL) meaning the load will fill up the entire truck, it is economical to ship directly from supplier to customer. Although when the products being shipped are less than a truckload (LTL), the other three strategies can be applied based on condition of the network (Dua et al., 2007).

Cross-docking approach is recognized as one of the basic distribution strategies which refers to a process, in which the products from different suppliers are collected (pickup process) and received at a cross-docking terminal, consolidated with other products shipped to the same destination without permanent storage and finally delivered to the final destinations (delivery process). The incoming shipments are unloaded at the inbound doors of cross-dock, sorted, consolidated and reloaded into outgoing vehicles within less than 36 hours. Other handling operations such as weighing, sizing, packaging, pricing and labeling products can also be done on shipments.

In some researches especially in vehicle routing problem with cross-docking (VRPCD) more complex cross-docking systems are referred, although the basic concept of cross-docking goes thorough one of cases demonstrated in Figure 1.

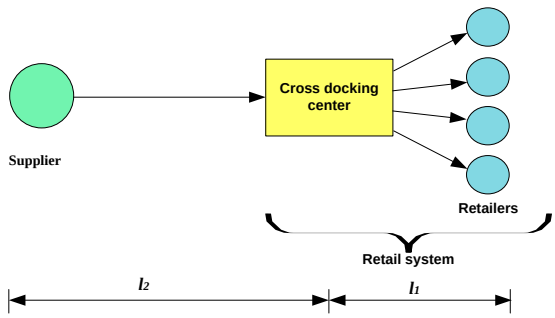


Figure 1-a) cross-docking network containing one suppler and many retailers ($l_2 \gg l_1$)

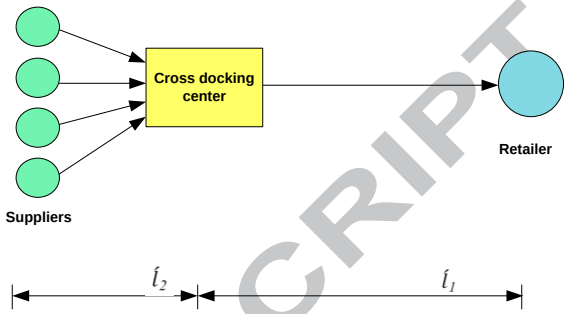


Figure 1-b) cross-docking network containing many suppliers and a retailer ($l'_1 \gg l'_2$)

As demonstrated in Figure 1-a, the network consists of one overseas suppliers and multiple retailers. With a cross-dock located in a region close to retailers, products from supplier are loaded to inbound vehicles, conveyed to the cross-dock, then sorted and loaded to outbound vehicles and finally delivered to retailers. The distribution system in Figure 1-b includes multiple suppliers and one retailer (customer) that orders to suppliers located in a long distance. It is more economical to consolidate products from various suppliers at a cross-dock located in their nearby, instead of direct transportation from each supplier to customer (Shaolong, 2007).

There are various extensions on two basic concepts mentioned above which one of them is shown in Figure 2.

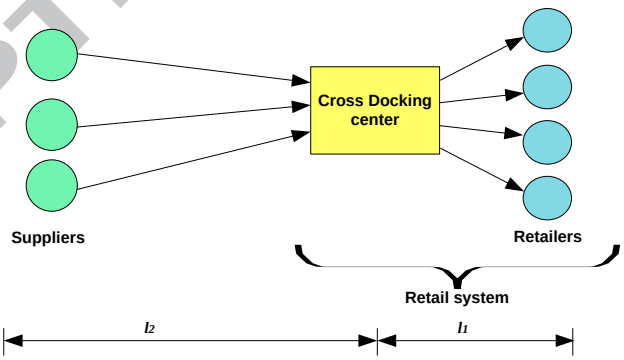


Figure 2. Cross-docking network containing multiple suppliers and multiple retailers (Shaolong, 2007)

This approach originates many benefits including the reduction of inventory holding and activities associated with storage of products; reduction of transportation cost by using full truckload and finally faster product flow in the network which leads to decreasing lead-times and improving customer service. Despite its advantages, employing this approach should be carefully evaluated since sometimes it is better to directly move shipments from suppliers to customers (Cóccola et al., 2015).

Decision problems in cross-docking strategy can be divided in some categories regarding the decision making level, i.e., strategic, tactical or operational (Buijs et al., 2014). The strategic level of decision making includes cross-docking location and their layout. In tactical level the problem of cross-docking network design is considered. The major decisions at the

operational level are the vehicle routing and scheduling, the dock door assignment and the truck scheduling at the cross-dock (Dondo and Cerda, 2015). Therefore, planning a cross-docking strategy involves different issues including location of cross-docks, vehicle routing, etc. that can be integrated in a problem. In addition, determining the transshipment strategy between suppliers and assembly plants will deeply affect the design of distribution system. In order to explain more, it is possible to directly shipped products from suppliers to plants and these flows should not have any effect on the location of cross-docks. A real application for this problem can be found in automotive industries in which different parts from various suppliers with a vast geographical distribution should be delivered to one or more assembly / production plants. Some suppliers located near plants or their load is near to full truckload, therefore in such situations direct shipment is more economical. For other suppliers may need to establish a (some) cross docking center(s).

Cross docking is a practical tool for implementing Just-in-time (JIT) delivery in supply chain operations. In the other hand, the main objective of JIT is to minimize transportation and inventory costs while delivering parts frequently and in small quantities. Frequent deliveries would be made in less than truckload (LTL) shipments that result in a considerable transportation cost. Accordingly, the consolidation of these small shipments into full truckloads (FTL) is more economical and involves routing problem to visit multiple suppliers in a route. The consolidation is mostly done in a cross-docking facility (Chuah, 2004).

In recent years, Just-in-time has become a viable, practical method especially in automotive industry, so the goal is to keep inventory levels down by shipping loads more frequently in less order volumes (Hosseini et al., 2014).

In order to achieve this goal, a model is presented that simultaneously considers location and vehicle routing problem and decides how the loads from each supplier to each assembly plant should be shipped. In fact, it allows less than truck loads (LTL) to be consolidated through cross-docking and allows high-volume loads to be shipped directly from suppliers to assembly plants.

The remainder of the paper is organized as follows. A review on the literature is presented in the next section. Section 3 gives the problem definition and presents a mathematical formulation. The solving methodology is presented in Section 4 in details and Section 5 devoted to the computational results. Finally, Section 6 concludes the paper and offers some suggestions for future research.

2. Literature

So far, some research has been done in various categories of cross-docking, which most of them considered scheduling of vehicles in the cross-dock, dock door assignment and vehicle routing problem, although there are only a few studies that have regarded VRP and location of cross-docking, simultaneously.

The following papers focused on location problem in cross-docking: Bhaskaran (1992), Sung and Song (2003), Gümüs and Bookbinder (2004), Sung and Yang (2008), Jayaraman and Ross (2003) and Ross and Jayaraman (2008), Bachlaus et al. (2008) and Musa et al. (2010).

Musa et al. (2010) proposed a model in which vehicles are allowed to be routed either directly from suppliers to customers or indirectly through one of the cross-docks in the distribution network. Based on their model, Hosseini et al. (2014) addressed the transportation problem in a consolidation system and developed a new mathematical formulation for it. They assumed three transportation strategies to move goods from suppliers to customers including direct shipment, indirect shipment (through cross-dock) and milk run.

In order to solve the problem the authors developed a hybrid algorithm based on harmony search (HS) and simulated annealing (SA). The objective function tries to minimize the total transportation cost by decreasing the number of required vehicles in the network. Sadjadi et al. (2009) also considered milk run method and proposed a mixed integer programming model for a case study in automobile industry in Iran. The authors utilized genetic algorithm (GA) to find near-optimal solutions based on actual information.

Mousavi and Tavakkoli-Moghaddam (2013) and Mousavi et al. (2014) addressed strategic, tactical and operational decision levels by considering the location problem of multiple cross-docking facilities and vehicle routing scheduling problem. In Mousavi et al. (2014) study, the problem has been made more realistic by considering uncertainty in decision levels. They developed a two-phase mixed integer linear programming (MILP) formulation and incorporated two types of uncertainties into mathematical formulation by proposing a hybrid fuzzy possibilistic–stochastic solution approach.

Another research which considered more than one problem in a single model is Dondo, and Cerdá's (2014) that presented a mixed integer linear programming model for the scheduling of an individual cross-dock, routing vehicles in the pickup and delivery process and the dock door assignment at cross-docking center. They assumed that the cross-dock has multiple strip and stack dock doors but the number of doors can be lower than the number of pick up and delivery vehicles, which means that the dock doors are considered as scarce resources. The content of the temporary storage has been also covered in their model, in this way that, some loads can be temporarily stored in front of the stack doors waiting for the arrival of other outbound vehicles.

In Dondo and Cerda (2015) recent study, a heterogeneous vehicle routing problem has been considered that also covers the internal transportation through the cross-dock from strip to the stack dock doors. They also have assumed that the number of dock doors is less than vehicles, thus queues of vehicles waiting for loading or unloading goods are unavoidable. They developed an integrated model to determine the routing and scheduling of vehicles, the dock door assignment, the truck docking sequence and required travel time to transport the goods from strip to assigned stack dock doors. A branch-and-cut search was employed to find solutions.

Ahmadizar et al. (2015) developed a model to determine the routing of vehicles in the pickup and delivery process regarding the variant in product prices offered by different suppliers. The model assigns products to suppliers and cross-docks and also optimizes the routes and schedules of vehicles and the consolidation process so that, the purchasing, transportation and holding costs are minimized. A genetic algorithm (GA) hybridized with a local search procedure is proposed to solve the problem.

Most papers in cross-docking approach considers a single objective formulation, although Mohtashami et al. (2015) proposed a multi-objective mathematical model that tries to minimize the make-span, transportation cost and the number of vehicle trips. This model also allows direct shipment from suppliers to the customers in addition to travels via cross-docks. In order to solve the multi-objective model, two meta-heuristic algorithms including the non-dominated sorting genetic algorithm (NSGA-II) and the multi-objective particle swarm optimization (MOPSO) have been presented. Results of numerical examples demonstrated the superiority of NSGA-II over the MOPSO.

Morais et al. (2014) addressed the VRPCD regarding vehicle capacity and time window constraints. They adapted a constructive heuristic and six local search methods, and three iterated local search heuristics for the problem and claimed that it differs from the other approaches in the literature because it only explores the space of feasible solutions. The

results indicate that their proposed algorithm improves the best solution known for half of the benchmark instances in the literature.

Shahin Moghadam et al. (2014) considered a similar problem with the focus on vehicle routing and scheduling in a network consisting of suppliers, customers and a cross-docking center. The capacity limitation and time windows for visiting customers and suppliers have been addressed. In addition, splitting services is possible, which means that a customer can be visited more than one time by different vehicles. A mixed integer non-linear programming model is presented to the problem and in order to solve it a simulated annealing and a hybrid algorithm based on ant colony and simulated annealing are provided.

Direct shipment has recently attracted attention from researches who are working in VRPCD. Cóccola et al. (2015) asserted that sometimes it is better to transport requests directly from suppliers to customers, thus applying this approach should be evaluated meticulously. They considered an actual problem in which employing direct delivery (without using cross-docks) is also discussed. In order to find near optimal solutions, a methodology based on column generation is proposed, which is embedded into an incomplete branch-and-price tree.

Mokhtarinejad et al. (2015) addressed most strategic and operational decision levels in cross-docking and developed an integrated programming model for vehicle routing and scheduling problem, in which the direct shipment from suppliers to destinations is also possible. They applied a machine-learning-based heuristic method (MLBM) to solve the model and used a clustering approach to group the customers, suppliers and locations of the cross-docks. In the other word, location and allocation to cross-docking centers are performed by the clustering approach. For routing vehicles and scheduling the visiting nodes, the authors formulated a TSP problem with two objective functions. In order to solve dock door assignment problem and schedule vehicles in the cross-docking centers a genetic algorithm has been proposed.

In order to study more in this scope, the reader can find more information on Boysen and Fliedner (2010); Agustina et al. (2010); Van Belle et al. (2012); and Buijs et al. (2014) investigations.

Going through the literature indicates that the location and vehicle routing problems in cross-docking need some adoptions to be applied in real cases. In distribution networks with cross-docking, moving shipments through cross-dock and consolidation process should be done in reasonable situations. Some real examples can be found in auto industries, chain stores, flower industry, etc. in which the efficient distribution strategy should be determined considering many factors. Thus, in this study, some of these considerations have been taken in to account to specify the best way to load and route the vehicles in the network.

From the other hand addressing studies in LRP area and comparing them with this study indicates that this study is the first one that applies location routing problem in cross docking to the best of author's knowledge. In the other word, it uses assumption of cross docking strategy and integrates them into location routing problem. Some issues that are not generally considered in LRPs but are addressed in this study are as follows:

- The consolidation process in cross-docks is regarded, since assembly plants may require various products from different suppliers. Products in various locations are collected in the cross-dock prior to transportation to their final destination. The inbound vehicles are reloaded at cross-dock and after classifying products according to their destinations, products are shipped from the cross-dock to their final locations via one or more vehicles. Consolidation is possible and economical for less than truckloads.
- The problem tries to minimize the number of used vehicles since less than truckload (LTL) is probable because of the just-in-time (JIT) environment in which the problem is considered. So the number of vehicles should be minimized.

- Supplier's products may be in different sizes that have effect on the number of required vehicles.

Moreover the problem does not limit the transportation strategy to cross docking and let another strategy (direct transportation from suppliers to assembly plant) to be used when it is appropriate. Based on this assumption many extensions to mathematical models of previous studies are needed. Studies in LRP field do not assume the direct shipment from factory to customers and all shipments pass through transfer points. Good literature reviews in LRP can be found in Nagy and Salhi (2007) and Prodhon and Prins (2014) papers.

3. Problem definition

The distribution network investigated in this study includes part suppliers (L), cross-docking centers (I), and assembly plants (J) (briefly called as plants). The transportation systems for shipping loads from suppliers to plants allow direct and indirect shipments (via cross-docks) in a way that the total transportation cost and the cost of establishing and utilizing cross-docking centers are minimized. We also assume that routing in pick up process is allowable for shipments moving through cross-docks.

Consolidation process is neither required nor possible for a FTL shipment which means that for quantity shipped equals (a multiple of) vehicle capacity, direct shipment is the most efficient way of transportation. The assembly plants may require different parts and products provided by several suppliers, meaning that multi product distribution network is considered. All assembly plants are directly connected to one or more cross-docks. Figure 3 shows the flow of shipments in a typical cross-docking network.

As can be inferred from Figure 3, there is no route between assembly plants. In order to explain this assumption more, consider a manufacturing group in which parts are usually shipped from suppliers to cross-docks by smaller vehicles such as pickup trucks, and from cross-dock to assembly plant by trailers. In a trailer generally the requests of a typical assembly plant is loaded and it is not economical to have routes between different plants because of their geographical distribution. In such situations, usually the number of plants is not great compared with the number of suppliers.

This paper develops a location routing problem (LRP) model based on Farahani and Hekmatfar (2009) under assumptions as follows:

- Every supplier sends products directly to assembly plants or thorough a cross-docking center.
- It is possible to have routes in the pick up process which means that it is allowable to have routes between suppliers, but every route starts and ends at a cross-dock.
- The load to be sent from each supplier to each assembly plant is known (d_{jl}). If d_{jl} is more than vehicle capacity, the solution is trivial and the vehicle is needed to go directly for that flow (from supplier l to plant j) since consolidation is not possible in such a situation due to full truckload and it is more economical to go directly to destination.
- There are multiple cross-docks with limited capacity in the distribution network.
- The vehicle fleet is homogeneous and all vehicles have the same capacity.
- The total load shipped by each vehicle cannot exceed the capacity.
- The route duration meaning the working time of each vehicle is limited.
- Vehicle route length is bounded by a given distance.
- The service time at each supplier location is known.

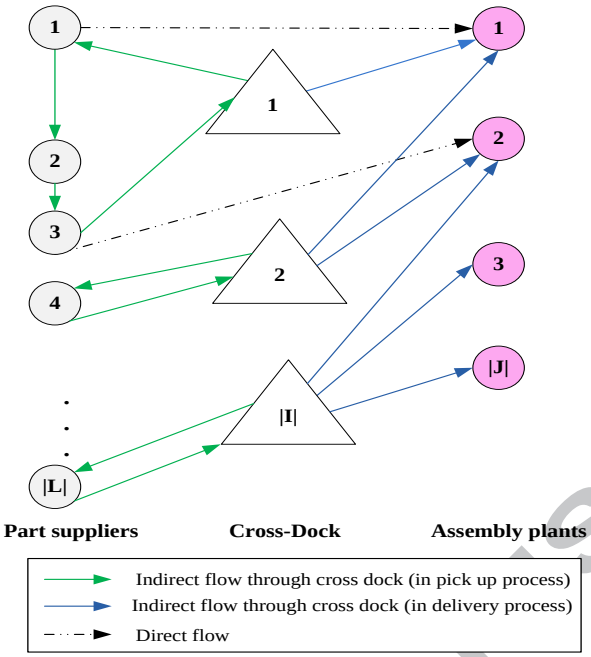


Figure 3. Considered cross-docking network when direct flow of products is allowable

- Each supplier is visited only once by a single vehicle in a tour in indirect shipment. In this transportation strategy, each route passes through only one cross-dock.
- The shipped parts are allowed to occupy various spaces of the vehicle.

The sets, parameters and variables used in the model are as follows:

Sets	
J	The set of assembly plants $\{j=1, 2, \ldots, J\}$
I	The set of cross-docks $\{i=1, 2, \ldots, I\}$
L	The set of parts suppliers $\{l=1, 2, \ldots, L\}$
K	The set of vehicles $\{k=1, 2, \ldots, K\}$

Parameters

c_{ij}	The distance between node i and j ($i, j \in (L \cup I \cup J)$)
p	The cost per distance unit
CO	Operational cost of vehicles
t_{ilk}	The travel time between node i and l by vehicle k
g_i	The fixed cost of establishing cross-dock i
v_i	The variable cost per commodity unit at cross-dock i
V_i	The maximum capacity of cross-dock i
d_{jl}	Demand of plant j from supplier l
s_{lk}	The time required by vehicle k to load at suppliers l
Q	The capacity of vehicles in number of standard units
su_l	The number of standard units per one unit of part produced by supplier l in terms of space needed
E_k	The maximum allowable length (in distance units) of vehicle k
T_k	The maximum allowable duration (in time units) of vehicle k

q_{il}	A fixed cost incurred in servicing by cross-dock i to supplier l
S	Arbitrary subset of set $(I \cup L)$

Decision variables

x_{lik}	1: if vehicle k moves from node l to the node i; 0: otherwise
y_i	1: if cross-dock i is open; 0: otherwise
U_{lj}	1: if load from supplier l to plant j is sent directly; 0: otherwise
Z_{il}	1: if load from supplier l is served by cross-dock i; 0: otherwise
F_{ij}	1: if plant j is served by cross-dock i; 0: otherwise
R_{lij}	1: if demand of plant j from supplier l goes thorough cross-dock i; 0: otherwise
w_{ij}	Quantity of parts shipped from cross-dock i to plant j.

With the notations introduced above, the problem can be formulated as a mixed integer non-linear programming (MINLP) model as follows:

$$Min \sum_{l\in L}\sum_{j\in J}(p \times c_{lj} + CO) \times U_{lj} + \sum_{i\in I} g_i \times y_i + \sum_{i\in I}\sum_{j\in J}(p \times c_{ij} + CO) \times (\left\lceil \frac{w_{ij}}{Q} \right\rceil) \times F_{ij} + \sum_{i\in I}\sum_{l\in L}\{v_i \times \sum_{j\in J} d_{jl}\,(1 - U_{lj}) + q_{il}\} \times Z_{il} + \sum_{k\in K}\sum_{l\in(L\cup I)}\sum_{i\in(L\cup I)} p \times c_{li} \times x_{lik} + \sum_{k\in K}\sum_{l\in L}\sum_{i\in I} CO \times x_{ilk} \quad (1)$$

S.t.

$$U_{lj} + \sum_{i\in I} R_{lij} = \min\{1, d_{jl}\};\ \forall\, l, j \quad (2)$$

$$\sum_{k\in K}\sum_{i\in(I\cup L)} x_{ilk} \le 1\ ;\ \forall\, l \quad (3)$$

$$\sum_{k\in K}\sum_{i\in(I\cup L)} x_{lik} \le 1\ ;\ \forall\, l \quad (4)$$

$$\sum_{l\in L}\sum_{i\in I} x_{lik} \le 1\ ;\ \forall\, k \quad (5)$$

$$\sum_{i\in I} Z_{il} \le 1\ ;\ \forall\, l \quad (6)$$

$$\sum_{l\in L} d_{jl}\,(1 - U_{lj}) \times Z_{il} - F_{ij} \times w_{ij} = 0\ ;\ \forall\, i, j \quad (7)$$

$$\sum_{j\in J} w_{ij} \le y_i \times V_i;\ \forall\, i \quad (8)$$

$$\sum_{l\in L}\sum_{i\in(I\cup L)}\{\sum_{j\in J} d_{jl} \times su_l \times (1 - U_{lj})\} \times x_{lik} \le Q;\ \forall\, k \quad (9)$$

$$\sum_{l\in(I\cup L)}\sum_{i\in(I\cup L)} c_{li} x_{lik} \le E_k;\ \forall\, k \quad (10)$$

$$\sum_{l\in(I\cup L)}\sum_{i\in(I\cup L)} t_{ilk}x_{ilk} + \sum_{l\in L}\sum_{i\in(I\cup L)} s_{lk}x_{ilk} \le T_k;\ \forall\, k \tag{11}$$

$$\sum_{l\in L} x_{lik} - \sum_{l\in L} x_{ilk} = 0\ ;\ \forall\, k,i \tag{12}$$

$$\sum_{u\in(L\cup I)} (x_{ulk} + x_{iuk}) - Z_{il} \le 1;\ \forall\, i,l,k \tag{13}$$

$$Z_{il} \le \sum_{j\in I} R_{lij}\ ;\ \forall\, i,l \tag{14}$$

$$\sum_{j\in I} R_{lij} \le |J| \times Z_{il};\ \forall\, i,l \tag{15}$$

$$F_{ij} \le \sum_{l\in L} R_{lij}\ ;\ \forall\, i,j \tag{16}$$

$$\sum_{l\in L} R_{lij} \le |L| \times F_{ij};\ \forall\, i,j \tag{17}$$

$$\sum_{k\in K}\sum_{l\in S}\sum_{i\in(I\cup L)-S} x_{lik} \ge 1; (2 \le |S| \le |I\cup L|;\ S \subseteq |I\cup L|;\ S\cap L \ne 0 \tag{18}$$

$$x_{lik} \in \{0,1\}\ ;\ (l\in L;\ i\in I; k\in K) \tag{19}$$

$$y_i \in \{0,1\};\ (i\in I) \tag{20}$$

$$U_{lj} \in \{0,1\};\ (l\in L;\ j\in J) \tag{21}$$

$$R_{lij} \in \{0,1\}\ ;\ (l\in L;\ i\in I; j\in J) \tag{22}$$

$$Z_{il} \in \{0,1\};\ (l\in L;\ i\in I) \tag{23}$$

$$F_{ij} \in \{0,1\};\ (i\in I;\ j\in J) \tag{24}$$

$$w_{ij} \ge 0;\ (i\in I;\ j\in J) \tag{25}$$

The objective function tries to minimize the total cost including direct transshipment cost, cross-dock fixed cost, delivery cost to assembly plants, variable cross-docking cost at receiving process, routing cost in pick up loads from suppliers and operational cost of utilized vehicles. Constraint (2) ensures that demand of plants from suppliers can be shipped via one of the two available transportation strategies.

According to Constraints (3) and (4), every supplier can be served at most in a route, which means that splitting services between vehicles is not allowed. Actually, Constraints (3) and (4) ensure that at most one vehicle can be enter and leave a supplier, respectively. Constraint (5) stipulates that every vehicle can only leave a cross-docking center. This prevents cases in which a vehicle departs the cross-dock for more than one supplier and also passes through two cross-docks. In Constraint (6), each supplier is served at most by one cross-docking center. Equation (7) determines the load to be sent from cross-docks to assembly plants and also ensures that a flow entering a cross-dock is equal to the flow exiting it.
Constraint (8) limits the flow through a cross-dock to the capacity of that cross-dock. Maximum vehicle capacity, maximum route length and maximum route duration are stipulated in Constraints (9-11). Constraint (12) guarantees the flow conservation equation: any point of ($L\cup I$) must be entered and left by the same vehicle. Constraint (13) ensures that Z_{il} be equal to 1, if supplier l and cross-dock i belong to the same route and cross-dock i serves supplier l.

Constraints (14) and (15) ensure that when $\sum_{j\in I} R_{lij}$ is positive, then Z_{il} should be equal to one, otherwise $Z_{il} = 0$. These constraints stipulate that cross-dock i gives service to supplier l, if R_{lij} is equal to one, at least for one assembly plant. Constraints (16) and (17) are similar to (14) and (15), but in delivery process.

Constraint (18) is the sub-tour elimination constraint guaranteeing that each tour must contain an open cross-dock from which it originates, i.e., each tour must consist of a cross-dock and some suppliers. The type of decision variables and the bound and integrality condition on them are defined in Constraints (19-25).

4. Solution methodology

As acclaimed by Nagy and Salhi (2007), LRP belongs to the class of NP-hard (Nondeterministic Polynomial-time hard) problems since it encompasses two NP-hard problems (facility location and vehicles routing). On the other hand, if cross-docks (distribution centers) are considered as constant and predetermined, LRP changes into VRP (Gharavani, Setak; 2015). Thus, if we assume the following assumption: the cross-docking centers as the depots (with characteristics of cross-docking networks), no direct shipments between each couple of nodes and one type of product in the transportation system, then the proposed model will be reduced to its basic (standard) form and it also will be an NP-hard problem. That is the reason why it is practically impossible to solve large size LRPs by exact methods in a reasonable amount of time. In order to solve large size problems, an algorithm based on Biogeography-based optimization (BBO) is proposed and its efficiency and performance is compared with particle swam optimization (PSO). BBO is a new meta-heuristic introduced by Simon (2008) to solve continuous optimization problems. Algorithm adaptation for discrete problems and steps of proposed algorithm are described in the bellowing sections.

4.1. Describing Biogeography-based optimization (BBO) algorithm

Biography based optimization (BBO) is a novel population-based algorithm inspired by natural ways of distributing species, the migration of species, and their extinction. The BBO components, their concepts and correspondence with optimization literature are reviewed in the following sentences (Simon, 2008; García-Torres et al., 2014):

- Habitat (H): In biography, it is the locality, site, residence or island occupied by biological species. It is analogous to a solution inside the search space of an optimization problem.
- Habitat Suitability Index (HSI): In biography, geographical areas that are well suited as residences for biological species are said to have a high HSI. So, HSI can be mapped to objective function from an optimization viewpoint. A good solution represents an island with a high HSI and a poor solution is a low HSI solution.
- Suitability Index Variables (SIV): SIV represents a feature of the solution (just like a gene in GA) (Rahmati and Zandieh, 2011). For more explanation, factors including land area, rainfall, diversity of vegetation and temperature effect on computation of the HIS values. Such factors are called as SIVs. SIV is used as a search variable, thus a set of all possible SIVs corresponds to the search space from which a solution is selected.
- Ecosystem: It refers to a group of N habitats and corresponds to the population of solutions from a population based optimization viewpoint.
- Immigration Rate: The control parameter λ is used to control habitat immigration. When a particular habitat is empty of any species, the maximum immigration rate (I) is realized in that habitat. As the number of species increases, it becomes more crowded and immigration rate decreases since fewer species are able to successfully survive in such crowd.

- Emigration Rate: Parameter μ controls habitat emigration. If there are no species in a habitat, emigration is equal to null. As the number of species increases, the habitat will be more crowded, leading to the fact that more species leave their habitat to explore other residences. Maximum emigration rate (E) in a single habitat occurs when it contains the largest number of species it can support. In optimization problems, HIS specifies λ and μ parameters.
- Migration model: There are various migration curves according to different mathematical models of biogeography (MacArthur & Wilson, 1967). In Figure 4, a simple linear model is demonstrated in which immigration and emigration curves are shown as straight lines. S_0 is the equilibrium number of species, at which the immigration and emigration rates are equal. I and E represent the maximum immigration and emigration rate, respectively and are mostly set to 1. S_{max} is the maximum number of species that the habitat can support (Simon, 2008).

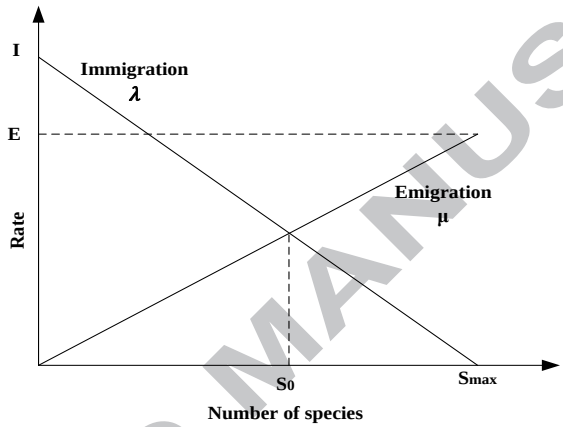


Figure 4. Linear model of migration (Simon, 2008)

As described by Simon (2008), the BBO can be informally explained in the following steps:

1) Initialize the BBO parameters (such as number of habitats, maximum migration and emigration rate, maximum mutation rate, S_{max}, etc.) and devising a method to map problem solutions to SIVs and habitats.
2) Generate a random set of habitats, each of which corresponds to a potential solution.
3) Map the HSI to the number of species S, the immigration rate and the emigration rate for each habitat.
4) Apply immigration and emigration strategies (See Section 4.2) to modify habitats then recalculate each HSI using the objective function.
5) Update the probability of containing S species for each habitat using Equation (28). Then, mutate habitats based on their probability as discussed in Section 4.2.3 and recalculate each HSI.
6) Repeat the algorithm from step (3) for the next iteration. This loop can be terminated after a predefined number of generations, or after achieving an acceptable solution to the problem.

4.2. Selection strategies

This step is what distinguishes BBO from other algorithms and executed through two main strategies: migration (also known as habitat modification) and mutation. The details of these strategies are explained as follows:

4.2.1. Migration strategy

Migration is a probabilistic operator that modifies each habitat H_i on the ecosystem H^n by sharing features among different habitat just like crossover operator of GA. Solutions are selected for immigrating or emigrating according to λ_i (immigration rate) and μ_j (emigration rate). Based on the concept of BBO, during the migration process two kinds of selection are addressed. First, it should be decided whether a particular habitat H_i should be immigrated or not. For this purpose, a random number is generated and simply compared with λ_i. Secondly, the habitat H_j should be chosen by using migration operator for emigrating to it regarding its emigration rate μ_j. The migration process can be demonstrated as: $H_i\,(SIV) \leftarrow H_j(SIV)$.

High HSI solutions (good solutions) tend to share their features with low HSI solutions (poor solutions) and emigrate to them. As mentioned before, by increasing the number of species, this tendency causes the immigration rate to decrease and the emigration rate to increase. Therefore, a high HSI solution is expected to have a relatively high μ and low λ, while a low HSI solution has a low μ and a high λ. After evaluating the HSI for each solution H_i, the immigration rate λ_i and emigration rate μ_j can be calculated via Equations (26) and (27), respectively.

$$\lambda_i = I(1 - \frac{k_i}{n}) \tag{26}$$

$$\mu_j = E(\frac{k_i}{n}) \tag{27}$$

Where k_i represents the rank of the i-th habitat after sorting all habitats according to their HSIs and n is the population size. Another parameter to be calculated is the probability that exactly S species exist in the habitat (P_S). This parameter changes from time to time as represented in Equation (28).

$$P_S(t + \Delta t) = P_S(t)(1 - \lambda_s \Delta t - \mu_s \Delta t) + P_{S-1}\lambda_{s-1}\Delta t + P_{S+1}\mu_{s+1}\Delta t \tag{28}$$

In order to model changes from time t to $t + \Delta t$, one of the following conditions may occur:

1) There are S species at time t and no immigration or emigration happened during $[t, t + \Delta t]$
2) There are S-1 species at time t and one species immigrated during $[t, t + \Delta t]$;
3) There are S+1 species at time t and one species emigrated during $[t, t + \Delta t]$.

4.2.2. Mutation strategy

Mutation is a probabilistic operator that randomly modifies the SIV of a solution based on a probability P_i (Simon, 2008). In BBO such as GA, this operator is applied to increase diversity among the population. The mutation probability m_i can be calculated through Equation (29) and according to solution probability. As can be inferred from Equation (29), mutation probability and solution probability are inversely proportional. In this equation, m_{max} is the maximum mutation rate that m can reach.

$$m_i = m_{max}(1 - \frac{p_i}{p_{max}}) \tag{29}$$

Now, regarding to the mutation probability m_i, the selection strategy and mutation operator can be implemented. General flowchart of the algorithm is presented in Figure 5. For more details, refer to Simon (2008).

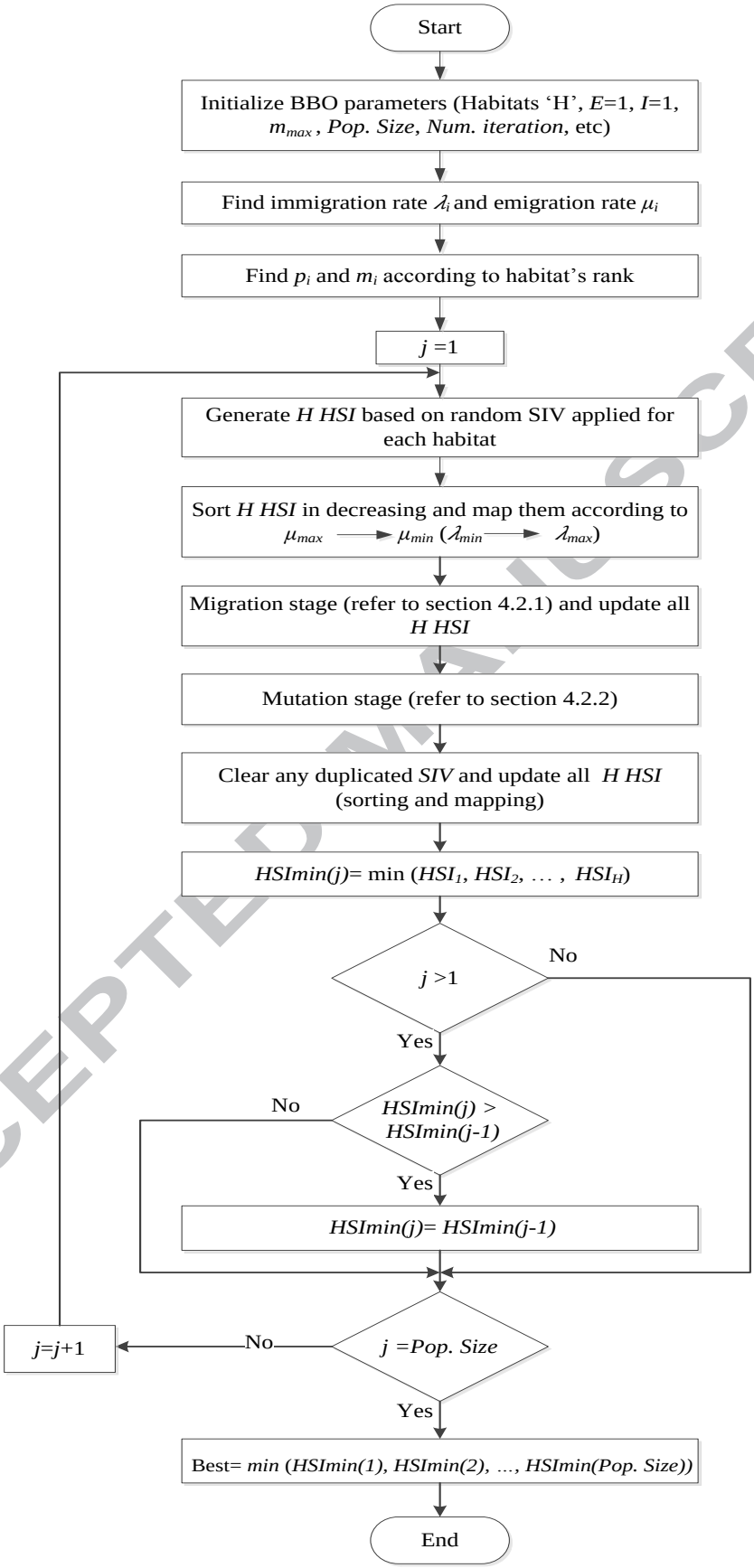


Figure 5. General flowchart of BBO algorithm (Alroomi et al., 2013)

4.3. Adaptation of algorithm for discrete problems

In order to use BBO algorithm to solve the proposed model that is a discrete one, first, some random real numbers between 0 and 1 are generated for suppliers and cross-docks. It should be noted that total number of nodes includes suppliers and cross-docks. For example, consider 7 suppliers and 3 cross-docks in the network. Numbers 1 to 7 are allocated to suppliers and cross-docks are corresponded to node 8, 9 and 10. We can specify the order of nodes by ascending sort of corresponding real numbers. Table 1 demonstrates this simple process of ordering, based on random real numbers in interval [0-1].

Table 1. An example of sorting nodes

Nodes (suppliers and cross-docks)	1	2	3	4	5	6	7	8	9	10
Random real numbers (in range [0-1])	0.32	0.69	0.83	0.14	0.19	0.03	0.55	0.52	0.91	0.37
Rank of real numbers	4	8	9	2	3	1	7	6	10	5
Order of nodes	6	4	5	1	10	8	7	2	3	9

In order to rank the nodes based on their corresponding real numbers, the least value is 0.03 which means that node 6 should be the first one in the string. Following is the order of nodes achieved by this process: {6-4-5-1-10-8-7-2-3-9} (the last row of Table 1).

4.4. Initialization of BBO algorithm and Representation of the Solution

Each solution consists of two parts: a L×J binary matrix and a string. The matrix specifies which loads to be sent directly (cells with 1's) and which loads pass through cross-dock (cells with 0's). In the above example, consider 3 assembly plants in the network. A random matrix is shown in Table 2.

Table 2. A binary matrix shows direct and indirect shipments

Suppliers \ Plants	1	2	3
1	1	0	0
2	0	0	1
3	1	0	1
4	1	1	0
5	0	1	1
6	0	1	0
7	1	1	0

Regarding this matrix, the load from each supplier to each plant has to be updated which means that just the loads related to cells with 0's will pass through cross-docks. For example, in Table 2, load from supplier 1 to plant 2 and 3, as well as from supplier 2 to plant 1 and 2, have to pass through cross-dock. Therefore, Table 2 is used to calculate remaining demand from each supplier that must be processed in indirect shipment.

The second part of solution just involves indirect loads and is a randomly generated permutation consists of all suppliers and cross-docks (see Figure 6).

6	4	5	1	10	8	7	2	3	9

Figure 6. Solution representation

In permutation, the position of nodes corresponds to cross-docks is a crucial issue. Nodes 8, 9 and 10 (cross-docks) are considered as delimiters which means that the substrings in their left side are allocated to them. For this example, suppliers {6-4-5-1} are allocated to node 10 and

suppliers {7-2-3} to node 9. Node 8 will not be established and is a closed cross-dock. Moreover, when a cross-dock is the first node in a string, that cross-dock will be considered as a closed one.

In the next step, vehicle / vehicles should be assigned to suppliers. In order to decide about vehicle routing, these items should be regarded: the remaining demand from suppliers, vehicles and cross-docks capacity as well as maximum allowable distance and duration (in time units) of vehicles. In this example, suppliers {6-4-5-1} will be serviced by a randomly chosen vehicle. The vehicle visits supplier 6 first (first ranked supplier), then if its capacity is not violated, goes to node 4 and this process continues until all assigned suppliers are visited or vehicle capacity or maximum allowable distance and duration is violated. In such condition, another vehicle is selected randomly and visits remaining suppliers. For substring {7-2-3}, a similar process will be done. For violating cross-dock capacity, a considerable penalty is used. Thus, opened cross-docks, allocation of suppliers to them and routing suppliers are specified in a string. Finally, Regarding to the demand of plants from suppliers and shipments that goes through a cross-dock, the quantity of parts shipped from each cross-dock to plants and subsequently number of vehicles needed from each cross-dock to each plant ($\left\lceil \frac{w_{ij}}{Q} \right\rceil$) will be determined.

4.5. Migration operator

As mentioned before, solution H_i is selected as immigrating habitat according to its immigration rate λ_i and H_j is chosen as emigrating habitat regarding its emigration rate μ_j. New position of H_i can be obtained using Equation (30) which is inspired by assimilation operator in the Imperialist competitive algorithm (ICA).

$$newposition_i = position_i + alpha \times (position_j - position_i) \tag{30}$$

where alpha is a parameter, which is used to implement migration process and is a number near to 1.

4.6. Mutation operator

In order to increase the diversity among the population, mutation operator is utilized. This operator is implemented by Equation (31):

$$newposition_i = newposition_i + sigma \times randn \tag{31}$$

where sigma is a parameter to change randomly the position of a habitat in mutation process and *randn* is a matrix of normally distributed random numbers.

The details of migration and mutation operators are devised based on strategies discussed in section 4.2. The pseudo code of algorithm is illustrated in Figure 7.

4.7. Particle swarm optimization

In order to compare BBO with another algorithm especially for large scaled problems, the particle swarm optimization (PSO) is utilized which is a solving algorithm widely used for Np-hard problems. According to Marinakis and Marinaki (2013), "*PSO is a very popular optimization method and its wide use, mainly during last years, is due to the number of advantages that this method has, compared to other optimization methods. Some of the key advantages are that this method does not need the calculation of derivatives, that the knowledge of good solution is retained by all particles and that particles in the swarm share information between them. … Concerning its implementation, PSO can easily be programmed, has few parameters to regulate and the assessment of the optimum is independent of the initial solution*". It should be noted that Marinakis and Marinaki's paper is in LRP filed.

```
Parameter setting: E=1, I=1, m_max=1, Pop. size, Num. iteration
Initialization: Generating habitats randomly as size as Pop. size
Population evaluation: evaluate habitats (just like chromosomes in GA)
Sort the population increasingly (from best to worst) based on HIS of habitats (cost)
For i=1: Num. iteration
    Calculate λ_i, μ_i, p_i and m_i according to habitat's rank
    For j=1: Pop. Size
        Generate Rand ϵ [0, 1]
        If Rand ≤ λ_j
          H_i(SIV)= select a habitat randomly through a Roulette wheel of μ (μ_1, μ_2, …, μ_n)
          Execute migration operator (like cross over operator of GA) (H_j (SIV), H_i(SIV))
        Else
          The habitat keeps unchanged
        End if
        Generate Rand ϵ [0, 1]
        If Rand ≤ m_j
          Execute mutation operator (like mutation operator of GA)
        Else
          The habitat keeps unchanged
        End if
    End for
    Calculate λ_j, μ_j, p_j and m_j according to habitat's rank
End for
```

Figure 7. Main algorithm of BBO (Rahmati & Zandieh, 2011)

PSO is a population-based algorithm proposed by Kennedy and Eberhart (1995) originated from swarm intelligence and social behavior of fish and birds. It has been observed that when birds seeking for food, the velocity of each member in a flock of them is affected by personal experience and information achieved from other members of the flock. This was the basic idea for PSO and it finds a solution for an optimization problem in a search space based on swarm intelligence. The PSO's solutions are as particles in the search space. Each particle has two main features: the position and velocity, which are clarified by both particles memory of their own best experience (the neighborhood's best position) noted as p_{best} and the global best of all members of the swarm known as g_{best}, which is immediately updated when a new best is found.

The velocity is updated by Equation (32), in which the first part represents the inertia of previous velocity and the second one, the 'cognition' part, refers to the private thinking by itself. The third part, the 'social' part, displays the cooperation between the particles (Kennedy 1997).

$$V_i(t+1) = w * V_i(t) + c_1 * r_1 * (x_{i-best}(t) \_x_i(t)) + c_2 * r_2 * (x_{gbest}(t) \_x_i(t)) \quad (32)$$

$$x_i(t+1) = x_i(t) + V_i(t+1) \quad (33)$$

Where $V_i(t)$, is the velocity of particle i in iteration t and $x_i(t)$ is the ith particle position. $x_{i-best}(t)$ is local best solution and shows the best previous position of each particle and $x_{gbest}(t)$ is the best position among all particles in the swarm. Inertial weight denoted as W, balances the global exploration and the local exploitation abilities of the swarm. c_1 and c_2 represent the weight of the stochastic acceleration terms. Finally, r_1 and r_2 are random functions to search better the space in the range [0, 1]. The inertial weight reduces slowly in each iteration and can be calculated by Equation (34):

$$w = w * wdamp \tag{34}$$

The V_{max} equation with search space width that has been presented by Shi and Eberhart (1999) is applied to converge more search space. As a result of using this approach particle's velocity and position must be limited through a defined space width.

4.8. Parameters tuning

As mentioned before, BBO and PSO are used to solve the problem. Meticulous tuning of their parameters will lead to achieve better solutions in efficient time. Thus, Taguchi method is utilized via Minitab software to design of experiments (DOE) and analyzing their results.

BBO parameters include number of habitats in the initial population (n_{pop}), number of kept habitats (*keeprate*), number of new habitats (n_{New}), mutation parameters ($p_{Mutation}$), and finally maximum emigration and immigration rates (E, I), whereas the parameters required for PSO are population size (*Swarm Size*), inertia weight (w), damping ratio (w_{damp}), personal learning coefficient (c_1) and global learning coefficient (c_2). In BBO, maximum emigration and immigration rates are set to 1 as mentioned in the literature. Moreover, *alpha* and *sigma* parameters are defined for migration and mutation processes, respectively. n_{pop} is considered as 180, 200 and 250 for small (problem set 1 and 2), medium and large size problems, respectively.

For each factor, four levels are defined and medium-size problems (with 20 suppliers) are solved for tuning parameters according to Taguchi method. Table 3 and 5 indicates the levels of BBO and PSO factors, respectively. The full factorial design requires L^m experiments, where m and L are the number of factors and levels, respectively. Therefor 4^4=256 experiments for parameters tuning of each algorithm is needed according to full factorial design, which is not efficient in terms of cost and time. Using "parameter design" developed by Taguchi in early 1960s, it is not needed to do experiments that cover all combinations of factors. In this method, controllable factors are placed in inner orthogonal array, while noise factors will be placed in the outer orthogonal array. Regarding to Taguchi method the appropriate array at least must have 16 rows (16 experiments) and the proper orthogonal array for both algorithms are L_{16} (4^4), which are indicated in Table 4 and 6 for BBO and PSO, respectively.

The objective function is considered as a quality measurement so minimum response value is preferred and its correspondent level of factors is known as appropriate level. After implementing experimental trials by using orthogonal array and calculating their means of means, the response value will be obtained. In the other word, the optimum condition for quality measurement is calculated by getting its response to different combinations of parameters. Thus, the optimal parameter level combinations can be determined.

In this approach, the algorithms are run 9 times with different values of above-mentioned parameters and their mean values are calculated. Then objective function is recorded in each run.

By analyzing diagrams, it can be found that which parameter value is appropriate and will lead to a better result. Obtained result corresponding to various parameters of BBO and PSO at different levels are depicted in Figures 8 and 9, respectively. Table 7 shows result of parameters tuning for both algorithms.

Table 3. Factor levels in BBO

factors	Index of levels	levels
keeprate	1	0.1
	2	0.2
	3	0.3
	4	0.4
alpha	1	0.8
	2	0.9
	3	0.95
	4	0.98
sigma	1	0.01
	2	0.02
	3	0.05
	4	0.1
$p_{mutation}$	1	0.1
	2	0.2
	3	0.3
	4	0.4

Table 4. The orthogonal array L_{16} (4^4) for BBO

Experiment	Parameters			
	Keeprate	*alpha*	*sigma*	$p_{mutation}$
1	0.1	0.8	0.01	0.1
2	0.1	0.9	0.02	0.2
3	0.1	0.95	0.05	0.3
4	0.1	0.98	0.1	0.4
5	0.2	0.8	0.02	0.3
6	0.2	0.9	0.01	0.4
7	0.2	0.95	0.1	0.1
8	0.2	0.98	0.05	0.2
9	0.3	0.8	0.05	0.4
10	0.3	0.9	0.1	0.3
11	0.3	0.95	0.01	0.2
12	0.3	0.98	0.02	0.1
13	0.4	0.8	0.1	0.2
14	0.4	0.9	0.05	0.1
15	0.4	0.95	0.02	0.4
16	0.4	0.98	0.01	0.3

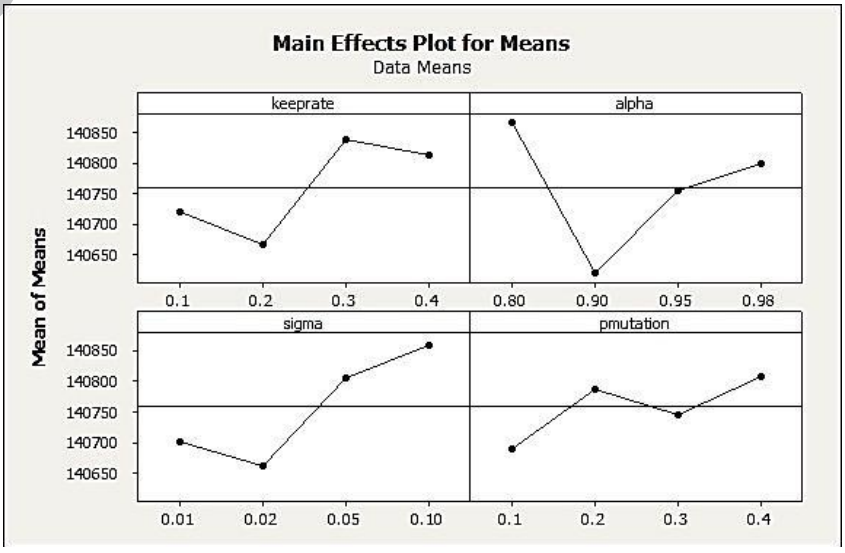


Figure 8. Analysis of Taguchi method for BBO parameters

Table 5. Factor levels in PSO

factors	Index of levels	levels
W	1	0.90
	2	0.95
	3	1.00
	4	1.10
w_{damp}	1	0.96
	2	0.97
	3	0.98
	4	0.99
c_1	1	1.0
	2	1.5
	3	2.0
	4	2.5
c_2	1	1.0
	2	1.5
	3	2.0
	4	2.5

Table 6. The orthogonal array $L_{16}(4^4)$ for PSO

Experiment	Parameters			
	w	w_{damp}	c_1	c_2
1	0.90	0.96	1.0	1.0
2	0.90	0.97	1.5	1.5
3	0.90	0.98	2.0	2.0
4	0.90	0.99	2.5	2.5
5	0.95	0.96	1.5	2.0
6	0.95	0.97	1.0	2.5
7	0.95	0.98	2.5	1.0
8	0.95	0.99	2.0	1.5
9	1.00	0.96	2.0	2.5
10	1.00	0.97	2.5	2.0
11	1.00	0.98	1.0	1.5
12	1.00	0.99	1.5	1.0
13	1.10	0.96	2.5	1.5
14	1.10	0.97	2.0	1.0
15	1.10	0.98	1.5	2.5
16	1.10	0.99	1.0	2.0

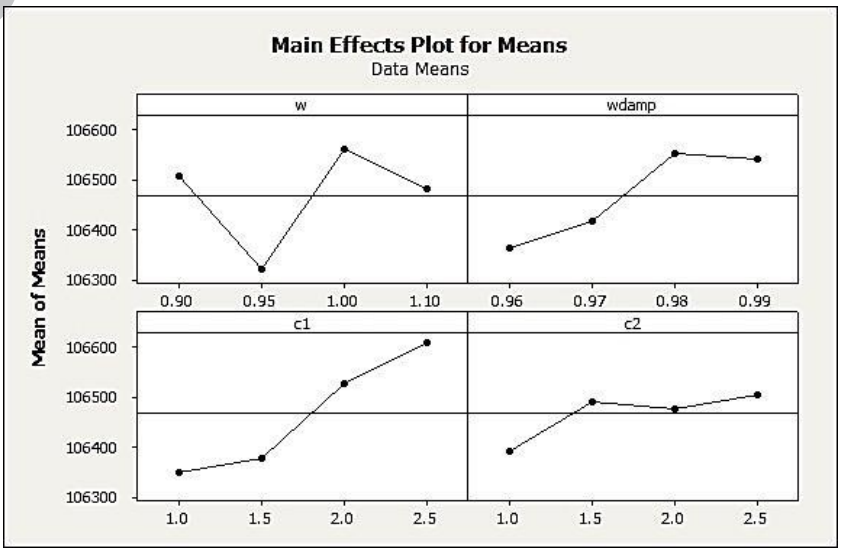


Figure 9. Analysis of Taguchi method for PSO parameters

Consequently, the level with the minimum value of response determines the optimum level of each factor. For example, in Figure 8, for *keeprate* factor, level 2 has the minimum value of objective function (when its value is set to 0.2). Similarly, the level 2 of *alpha* (*alpha*= 0.9) and level 2 of sigma (*sigma*=0.2) and level 1 of $p_{mutation}$ have also indicated the optimum situation in terms of mean value.

Table 7. The appropriate level for BBO and PSO parameters

Tuned Parameters for BBO			
Keeprate	**alpha**	**sigma**	**Pmutation**
0.2	0.9	0.02	0.1
Tuned Parameters for PSO			
w	$\mathbf{w_{damp}}$	$\mathbf{c_1}$	$\mathbf{c_2}$
0.95	0.96	1	1

5. Computational results

This section presents the results of computational study conducted to investigate the performance of the presented mathematical model and solution algorithms. First, the intervals used for generating problem instances are described. Then a sample instance is presented. The proposed model is solved on totally 20 + 1 problem instances using GAMS 24.2 software (for small-scale problem instances) and proposed BBO and PSO algosrithms. In addition, the number of applied vehicles is reported in comparison to the situation where just direct shipment is possible and there is no cross-dock in the network. Besides, the solutions are also compared with the results of condition in which all loads should pass through a cross-dock and direct shipping is not allowable. These were run on a PC with an Intel Core 5 Duo CPU (2.33 GHz) and 2 GB memory.

5.1. A sample instance

This section presents a simple instance of the problem considering 10 part suppliers, 3 candidate nodes to establishing cross-docks and 2 assembly plants. The written numbers under each supplier node indicates the demand of assembly plants from that supplier as is shown in Figure 10.

For this instance, the vehicle capacity is 150 and the x-y coordinate of all nodes is assumed to be in a plane of [0, 1]. The cost between each couple of nodes is supposed to be as distance between them (p=1). Number of standard units per one part unit in terms of space (Su) is set to 1.

In the studied addressed LRP so far, the researches assume that all loads should be sent through depots and direct shipment is not possible in those LRP's. While in this study direct shipping is also a strategy to meet the demands. An important point is that if fixed cost of establishing cross-docks is much more than other cost of transportation, the solution will be trivial, there should be no cross-dock in the network and all loads will go directly to assembly plants. Therefore, the establishing (commissioning) cost should be determined using a real approach. If we consider a depreciation period of 12 years for cross-docks and a cost of 10000 units to establish them, its cost will be about 833 units each year that should be divided by 12*15 since it is assumed that cross-docks are used about 15 times in a month. Thus, the cost of employing every cross-dock will be about 4.6 units for each time of usage. In this condition, the transportation cost between each couple of nodes is considered to be averagely 0.5 units and the relation between average fixed cost and average transportation cost is about 9.26. It should be noticed that using cross-docks will decrease the number of applied vehicles and their operational cost.

Regarding aforementioned reason, the establishing cost of cross-docks is selected in the interval [4, 9] and transportation cost between each pair of nodes follows a uniform distribution in (0, 1). Figure 10 illustrates the solution of this instance solved by BBO.

With running the algorithm for this instance, the results reached as follows: cross-dock 2 is the only opened cross-dock and 11 vehicles are used for direct shipment, while two vehicles are utilized for moving parts to cross-dock 2. In the delivery process, 186 and 103 parts are delivered to assembly plant 1 and 2, respectively, which means that totally three (two vehicles for plant 1 and one vehicle for plant 2) vehicles are applied in the delivery process. In order to avoid the complexity of figure, the delivery process is not indicated in Figure 10. If we consider a network in which just direct shipment strategy is possible, then number of required vehicles will be equal to 20 since each assembly plant needs parts from all suppliers, although by using cross-docking strategy in the distribution network, 16 vehicles are needed. The objective cost is 25.17. When there are more suppliers in the network, the total cost will decrease more in comparison with the situation in which just direct shipment is allowable.

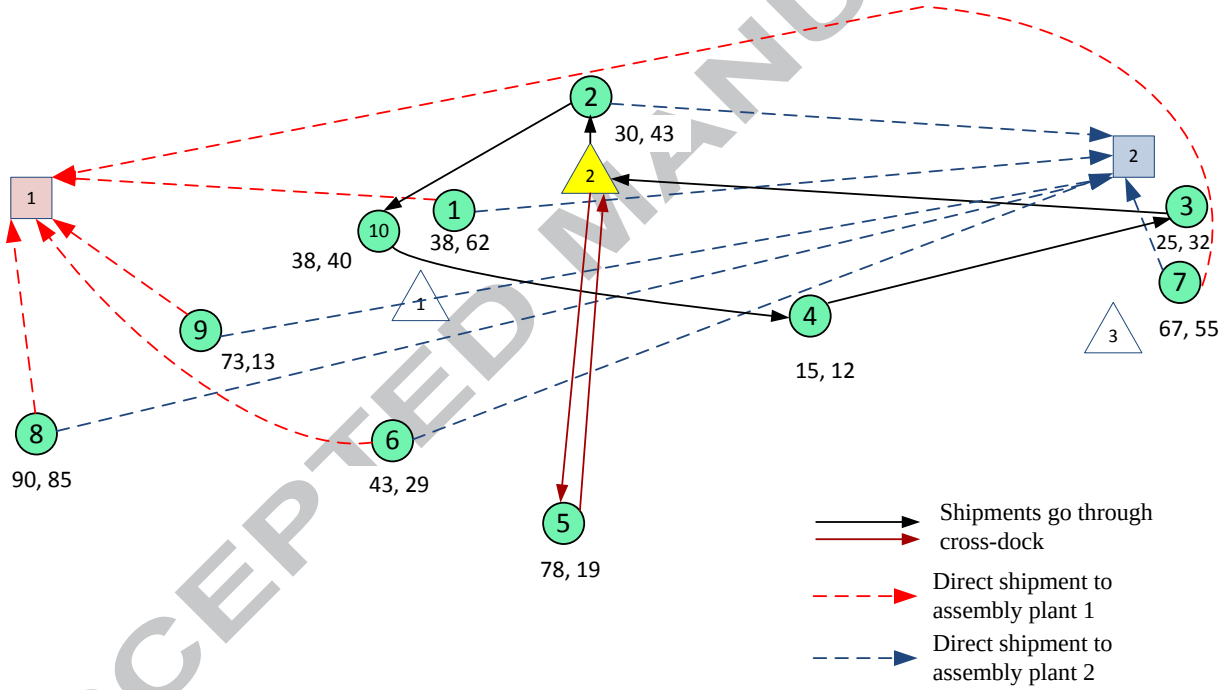


Figure 10. The solution of this instance solved by BBO

5.2. Experimental results

The problem sets, generated based on Hosseini et al. (2014), are assumed to be as follows: set 1(10 3 1) (i.e. 10 part suppliers, 3 cross-docks and 1 assembly plant), set 2 (17 5 2), set 3 (24 6 3), set 4 (30 8 4). The maximum capacity of vehicles is assumed to be 150 for set 1 and 2 and 300 for set 3 and 4. The maximum capacity of cross-docks is set to 900, 1500, 2000 and 2500 for set 1, 2, 3 and 4, respectively. The demand of plants is randomly generated in Uniform (20 and 90). Other characteristic and intervals of instances are presented in Table 8.

Table 8. The characteristic and intervals of instances

Parameters	Value and intervals
Transportation distance between each couple of nodes	U(0, 1)
Travel time (t_{ilk})	U(0.3, 1.8) Hour
Service Time (sl_k)	U(0.3, 0.8) Hour
Work time (maximum allowable duration) (T_k)	8 Hours
The maximum allowable distance of vehicles (E_k)	5
Fixed cost of establishing cross-docks	U(2, 5)
The variable cost per commodity unit at cross-dock i (v_i)	U(0.5, 1)
A fixed cost incurred in servicing by cross-dock i to supplier l (q_{il})	U(0.1, 0.5)
Number of standard units per one part unit produced by supplier l (Su_l)	U(0.5, 1.5)

As mentioned in section 5.1, the transportation cost between each couple of nodes is assumed to be as distance between them. Table 9 represents the result of solving model by BBO, PSO and GAMS software in terms of objective function and CPU time for problem set 1.

Table 9. Computational results for problem set 1

Problem instance	GAMS		PSO			BBO		
	Objective value	CPU (sec)	Objective value	CPU (sec)	Gap (%)	Objective value	CPU (sec)	Gap (%)
1-1	19.794	960	19.806	49.85	0.06	19.794	29.01	0.00
1-2	17.169	960	17.440	35.89	1.58	17.380	28.32	1.23
1-3	18.332	960	18.772	35.85	2.40	18.752	29.00	2.29
1-4	18.751	960	19.141	36.49	2.08	19.536	35.30	4.19
1-5	18.598	960	18.946	35.13	1.87	18.598	42.68	0.00
1-6	20.174	960	20.461	38.49	1.42	20.564	30.21	1.93
1-7	19.622	960	20.790	36.22	5.95	20.200	28.30	2.95
1-8	19.004	960	20.139	35.45	5.97	19.795	29.64	4.16
1-9	18.019	960	19.110	35.26	6.05	18.519	29.16	2.77
1-10	18.885	960	19.664	36.26	4.12	19.332	29.23	2.36

For problem set 1, it spends too much time to solve instances using GAMS software, so 16 min runs were used as limit for solving this problem set. By increasing the size of problem, GAMS software is not able to solve it because of the large number of parameters and constraints.
The column ‘‘Gap (%)” shows the gap between the result of solving model by GAMS and BBO and PSO methods. As it can be seen in Table 9, average Gap of PSO and BBO for problem set 1 are 3.15 and 2.19, respectively. Maximum gap of PSO and BBO are 6.05 and 4.19 for this problem set. It should be noted that computational time of two algorithms are considerably shorter than that of GAMS. More specifically, for problem set 1, PSO could find solutions, on average, in 37.49 Sec, which is about 4% of CPU time of GAMS and BBO solves problems in 31.08 Sec, which is about 3% of GAMS computational time.

GAMS results are not applicable for problem set 2 and more complicated problems, therefore solutions obtained by PSO and BBO are presented and compared to each other. BBO can achieve about 4.07%, 6.09% and 8.67% better solutions compared with PSO, for problem set 2, 3 and 4, respectively. Just in one case (instance 3-7) PSO outperforms this algorithm. Standard deviation of both algorithms for four problem sets is reported in Table 10, which is calculated based on computational results of problem set 1-4. The table shows that BBO solutions are more scattered compared with those of PSO.

Table 10. Comparing mean and standard deviation of objective values of algorithms

Problem set	PSO (Objective value)		BBO (Objective value)	
	mean	Standard Deviation	mean	Standard Deviation
set 1	19.427	0.918	19.247	0.895
set 2	48.805	2.960	46.902	2.716
set 3	97.499	4.543	92.084	5.599
set 4	168.318	5.916	155.032	6.715

In the following figures, two solution algorithms are compared in terms of objective function value (OFV) (Figure 11) and time criterion of algorithm runs (Figure 12). Horizontal axis shows the problem instance number.

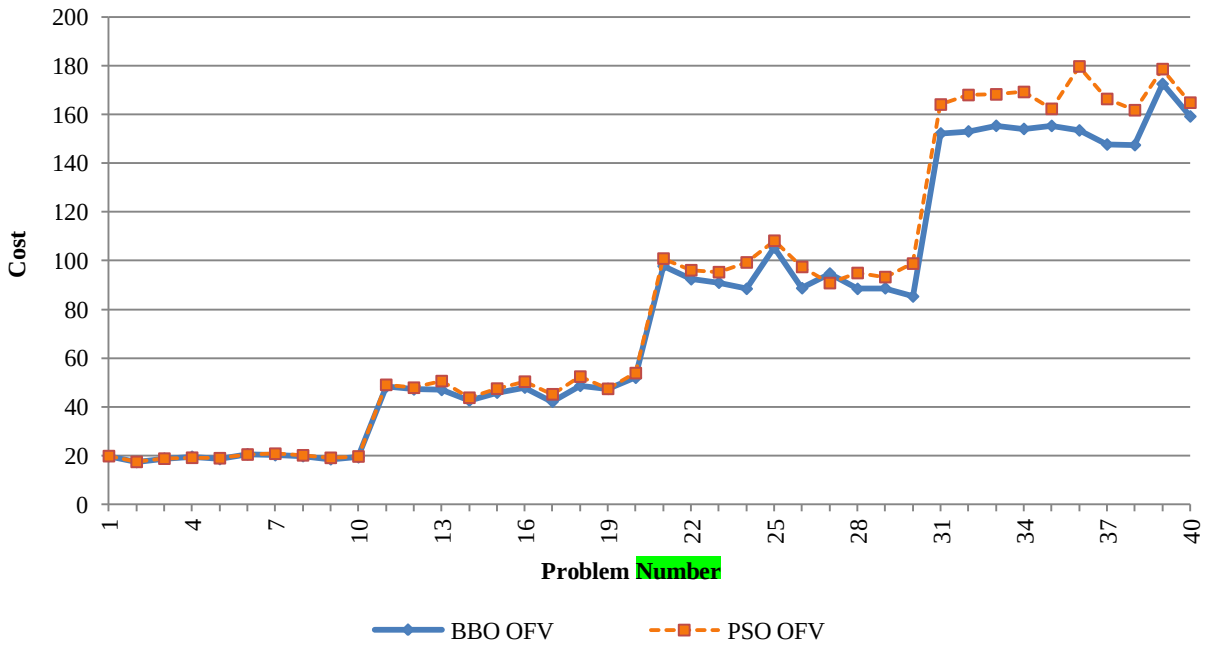


Figure 11. The comparison of algorithms based on OFV criteria

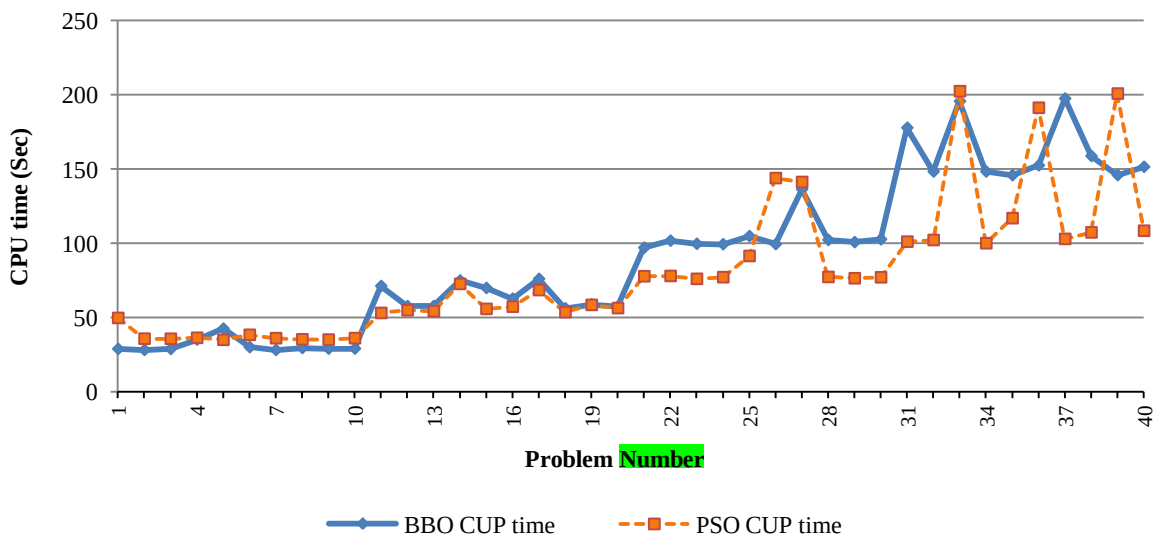


Figure 12. The comparison of algorithms based on the computational time

In brief, for small size problems, utilizing two algorithms will result in approximately the same solutions, although by increasing the problem size, the difference between results got larger and BBO reaches better solutions. As indicated in the Figure 12, in most instances PSO finds near optimal solutions in a less computational time, though both algorithms perform fast. Figure 13 illustrates the convergence process of BBO algorithm for test problem 4-3 with 30 suppliers, 8 cross-docks and 4 assembly plants.

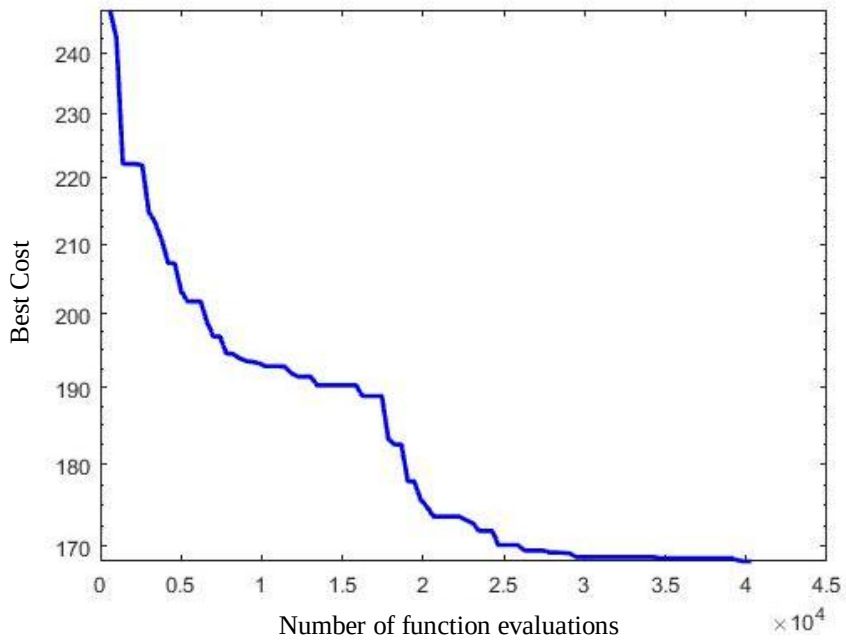


Figure 13. Convergence process of BBO algorithm for test problem 4-3

6. Conclusion and future research

Cross-docking is a logistic strategy to consolidate shipments from different suppliers and sort them for distribution to various customers. This study addressed a model for designing a consolidation network and transportation problem of such network in which direct shipping and cross-docking strategies are both possible to send the loads from suppliers to assembly plants. By considering two problems of location and distribution planning simultaneously, the costs of supplying parts may decrease since if some loads can be directly shipped, then they should not be taken into account in determining cross-docks location. Usual models for vehicle routing in cross-docking networks force all vehicles to go through cross-dock even if the shipment is near to full truckload (FTL); the vehicle collects and delivers the same set of products or when it is not economical to pass through a cross-dock because of the distance between the supplier and the customer.

In order to specify an economical and tailored network, we consider location and routing problems in an integrated model. In such systems, the model determines the optimal location of cross-docks, the loads to be sent directly from suppliers to plants and the loads to be sent indirectly (through cross-docks), vehicle routing and assignment of suppliers and assembly plants to cross-docks.

As mentioned in the literature, the model is NP- hard, thus two heuristic approaches based on BBO and PSO were proposed to solve it. The proposed algorithms and GAMS were tested on four problems sets (totally 40 instances) for verification. The computational results and comparisons of their outputs were also presented. BBO results outperformed PSO solutions

in terms of objective function especially for large problem instances. In terms of computational time, in most instances PSO is able to find solutions with less CPU time, while for problem set 1, running time of BBO is generally less than that of PSO.

The authors are looking forward to extending the current study by considering some main ideas and assumptions to make the problem more applicable for several real cases. Some of the main extensions are as follows.

In the mathematical model, some assumptions could be relaxed. As an example, it has been assumed that the loads to be sent from suppliers to plants are known in the current model, while the model could be formulated alternatively to determine the load amount to be transported from suppliers to plants according to their demand. Considering a fleet of vehicles with different capacities (heterogeneous fleet) is another relaxation of assumptions. Moreover, the best capacity of each cross-dock can be determined by the model.

Using milk run strategy in the distribution network can be another principal extension for the study in which the loads from some suppliers are delivered to a plant by the same vehicle. In addition, solving the transportation problem by considering time window in delivery and pickup process will lead to a more realistic problem. In such a problem, the total transportation time is minimized or the time horizon constraint of suppliers and plants is met.

References

[1] Agustina, D., Lee, C.K.M., & Piplani, R. (2010). A Review: Mathematical Models for Cross Docking Planning. *International Journal of Engineering Business Management*, 2, 47-54.

[2] Ahmadizar, F., Zeynivand, M., & Arkat, J. (2015). Two-level vehicle routing with cross-docking in a three-echelon supply chain: A genetic algorithm approach. *Applied Mathematical Modelling*, 39(22), 7065–7081.

[3] Alroomi, A. R., Albasri, F. A., & Talaq, J. H. (2013). Solving the associated weakness of biogeography-based optimization algorithm. *International Journal on Soft Computing*, 4, 1-20.

[4] Apte, U.M. & Viswanathan, S. (2000). Effective cross docking for improving distribution efficiencies. *International Journal of Logistics*, 3, 91–302.

[5] Bachlaus, M., Pandey, M.K., Mahajan, C., Shankar, R., & Tiwari, M.K. (2008). Designing an integrated multi-echelon agile supply chain network: a hybrid taguchi- particle swarm optimization approach. *Journal of Intelligent Manufacturing*, 19(6), 747–61.

[6] Bhaskaran, S. (1992). Identification of transshipment center locations. *European Journal of Operational Research*, 63(2), 141–150.

[7] Boysen, N. & Fliedner, M. (2010). Cross-dock scheduling: Classification, literature review and research agenda. *Omega*, 38, 413–422.

[8] Buijs, P., Vis, I. F.A., & Carlo, H. J. (2014).Synchronization in cross-docking networks: A research classification and framework. *European Journal of Operational Research*, 239, 593–608.

[9] Chopra, S.C., & Meindl, P. (2001). Supplier Chain Management – Strategies, Planning, and Operation. *Prentice Hall*, Upper Saddle River, New Jersey, 22–23.

[10] Cóccola, M., Méndez, C. A., & Dondo, R. G. (2015). A branch-and-price approach to evaluate the role of cross-docking operations in consolidated supply chains. *Computers and Chemical Engineering*, 80, 15–29.

[11] Dondo, R. & Cerdá, J. (2014). A monolithic approach to vehicle routing and operations scheduling of a cross-dock system with multiple dock doors. *Computers and Chemical Engineering*, 63, 184–205.

[12] Dondo, R. & Cerdá, J. (2015). The heterogeneous vehicle routing and truck scheduling problem in amulti-door cross-dock system. *Computers and Chemical Engineering*, 76, 42–62.

[13] Dua, T., Wang, F.K., & Lu, Pu-Yun. (2007). A real-time vehicle-dispatching system for consolidating milk runs. *Transportation Research Part E.*, 43, 565–577.

[14] Farahani, R. Z., & Hekmatfar, M. (2009). Facility Location: Concepts, Models, Algorithms and Case Studies. Springer.

[15] García –Torres, J.M., Damas, S., Cordon, O., & Santamaria, J. (2014). A case study of innovative population-based algorithms in 3D modeling: Artificial bee colony, biogeography-based optimization, harmony search. *Expert Systems with Applications*, 41, 1750–1762.

[16] Gharavani, M., & Setak, M. (2015). Capacitated Location Routing Problem with Semi Soft Time Windows. *Advanced Computational Techniques in Electromagnetics*, 1, 26-40.

[17] Gumus, M. & Bookbinder, J.H. (2004). Cross-docking and its implications in location-distribution systems. *Journal of Business Logistics*, 25(2), 199-228.

[18] Hosseini, S. D., Akbarpour Shirazi, M., & Karimi, B. (2014). Cross-docking and milk run logistics in a consolidation network: A hybrid of harmony search and simulated annealing approach. *Journal of Manufacturing Systems*, 33, 567–577.

[19] Jayaraman, V., & Ross, A. (2003). A simulated annealing methodology to distribution network design and management. *European Journal of Operational Research*, 144, 629–645.

[20] Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *In Proceeding of the 1995 IEEE international conference on neural network*, Perth, Australia, pp. 1942–1948.

[21] Kennedy, J. (1997). The particle swarm: social adaptation of knowledge. *Proc. Intl. Conf. on Evolutionary Computation*, India Service Center.

[22] MacArthur, R., & Wilson, E. (1967). The theory of biogeography. *Princeton University Press*, Princeton.

[23] Marinakis, Y., Marinaki, M. (2013). A Bilevel Particle Swarm Optimization Algorithm for Supply Chain Management Problems. Book chapter: Metaheuristics for Bi-level Optimization, Studies in Computational Intelligence, 482, 69-93.

[24] Mohtashami, A., Tavanab, M., Santos-Arteagad, F. J., & Fallahian-Najafabadi, A. (2015). A novel multi-objective meta-heuristic model for solving cross-docking scheduling problems. *Applied Soft Computing*, 31, 30–47.

[25] Mokhtarinejad, M., Ahmadi, A., Karimi, B., & Rahmati, S. H. A. (2015). A novel learning based approach for a new integrated location-routing and scheduling problem within cross-docking considering direct shipment. *Applied Soft Computing*, 34, 274–285.

[26] Morais, V.W.C., Mateus, G.R., & Noronha T. F. (2014). Iterated local search heuristics for the Vehicle Routing Problem with Cross-Docking. *Expert Systems with Applications*, 41, 7495–7506.

[27] Mousavi, S.M. Vahdani, B., Tavakkoli-Moghaddam, R., & Hashemi, H. (2014). Location of cross-docking centers and vehicle routing scheduling under uncertainty: A fuzzy possibilistic–stochastic programming model. *Applied Mathematical Modelling*, 38, 2249–2264.

[28] Mousavi, S.M., & Tavakkoli-Moghaddam, R. (2013). A hybrid simulated annealing algorithm for location and routing scheduling Problems with cross-docking in the supply chain. *Journal of Manufacturing Systems*, 32, 335–347.

[29] Musa, R., Arnaout, J.P. & Jung, H. (2010). Ant colony optimization algorithm to solve for the transportation problem of cross-docking network. *Computers & Industrial Engineering*, 59, 85-92.

[30] Nagy, G., & Salhi, S. (2007). Location-routing: issues, models and methods. *European Journal of Operational Research*, 177, 649–672.

[31] Prodhon, C. & Prins, C. (2014). A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238, 1–17.

[32] Rahmati, S. H., & Zandieh, M. (2011). A new biogeography-based optimization (BBO) algorithm for the flexible job shop scheduling problem. *Int J Adv Manuf Technol*, 58, 1115-1129.

[33] Ross, A., & Jayaraman, V. (2008). An evaluation of new heuristics for the location of cross-docks distribution centers in supply chain network design. *Computers & Industrial Engineering*, 55, 64–79.

[34] Sadjadi, S. J., Jafari, M., & Amini, T. (2009). A new mathematical modeling and a genetic algorithm search for milk run problem (an auto industry supply chain case study). *Int J Adv Manuf Technol*, 44, 194–200.

[35] Shahin Moghadam, S., Fatemi Ghomi, S.M.T., & Karimi, B. (2014). Vehicle routing scheduling problem with cross docking and split deliveries. *Computers and Chemical Engineering*, 69, 98–107.

[36] Shi, Y., & Eberhart, R. C. (1999). Empirical study of particle swarm optimization. P*roceedings of the 1999 Congress on Evolutionary Computation*, pp. 1945-1950. Piscataway, NJ: IEEE Service Center.

[37] Simon, D. (2008). Biogeography-based optimization. IEEE Trans Evol Comput 12, 702–713.

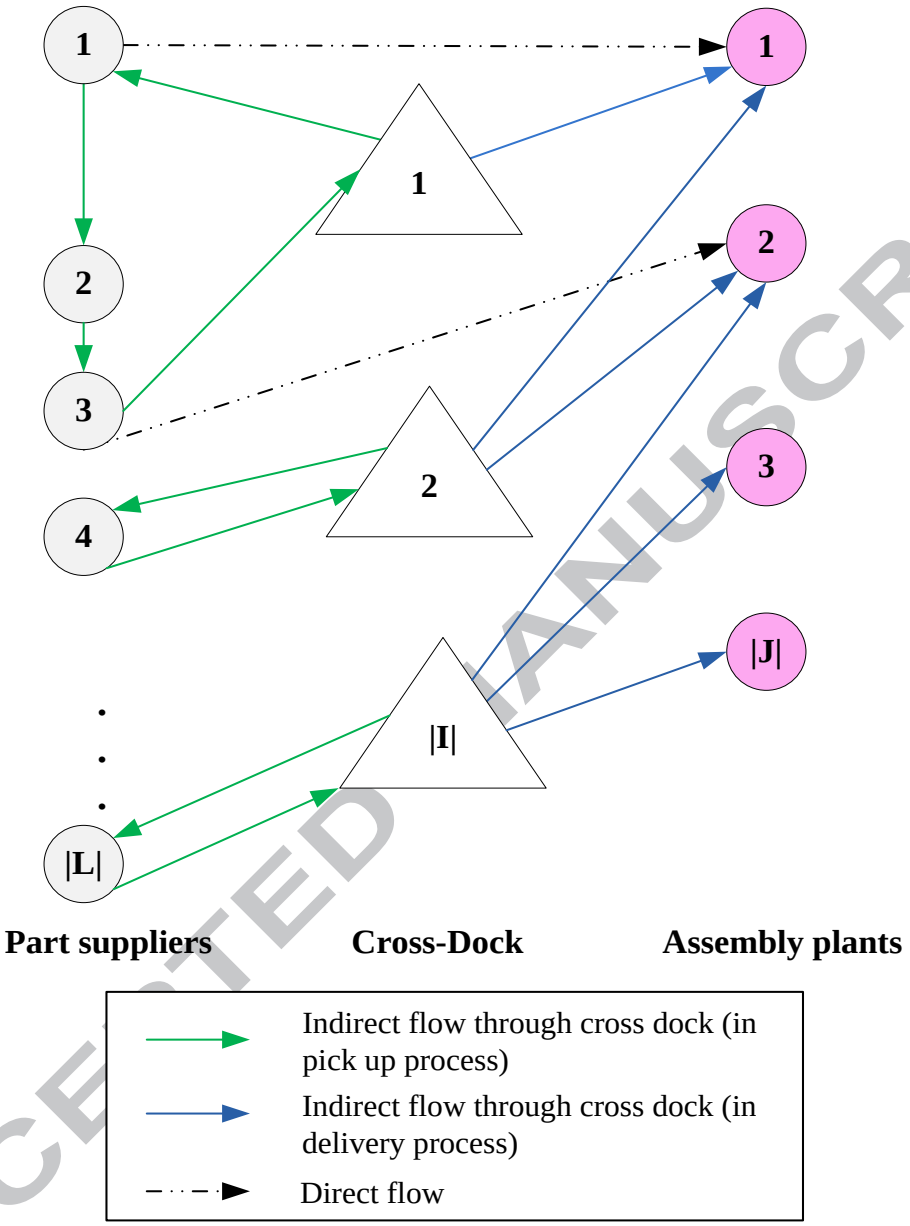
[38] Shaolong, T. (2007). the impact of cross-docking operation on supply chain management. *A thesis for Degree of philosophy*.

[39] Sung, C.S., & Song, S.H. (2003).Integrated service network design for a cross-docking supply chain network. *The Journal of the Operational Research Society*, 54(12), 1283-1295.

[40] Sung, C.S., & Yang, W. (2008). An exact algorithm for a cross-docking supply chain network design problem. *Journal of the Operational Research Society*, 59(1), 119–36.

[41] Van Belle, J., Valckenaers, P. & Cattrysse, D. (2012). Cross-docking: State of the art. *Omega*, 40, 827–846.

Graphical abstract



Highlights

- An integrated model for location-routing problem in a distribution network with cross-docking centers is proposed.
- In the distribution network, loads can be transported via direct shipment in addition to cross-docking strategy.
- In cross docking it is possible to have routes between suppliers.
- A metaheuristic algorithm based on Biogeography-based optimization (BBO) is proposed.