Malaysian Journal of Mathematical Sciences 15(1): 1–20 (2021)

MALAYSIAN JOURNAL OF MATHEMATICAL SCIENCES

Journal homepage: http://einspem.upm.edu.my/journal

Modified Harmony Search Algorithm for Location-Inventory-Routing Problem in Supply Chain Network Design with Product Returns

Misni, F. [2,3] and Lee, L. S. * [1,2]

[1] *Laboratory of Computational Statistics and Operations Research, Institute for Mathematical Research, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia*
[2] *Department of Mathematics, Faculty of Science, Universiti Putra Malaysia, 43400 UPM Serdang, Selangor, Malaysia*
[3] *Centre for Mathematical Sciences, Universiti Malaysia Pahang, 26300 Gambang, Kuantan, Pahang, Malaysia*

E-mail: lls@upm.edu.my
* *Corresponding author*

Received: 30 March 2020
Accepted: 30 December 2020

ABSTRACT

This paper presents a study on the integration of location-inventory-routing problem by considering the returned products of an economic production quantity model in supply chain network design. This model assumes that the returned products can be re-furbished and re-entered into the market to be re-sold. The problem aims to minimise the total cost of establishing the depots, the cost of setup the production and inventory holding, and the cost of delivery travelled by the vehicles. This problem is solved by a proposed Modified Harmony Search (MHS) algorithm with dynamic parameter setting and multi-local neighbourhood search techniques. Computational experiments on benchmark instances show that the proposed MHS outperformed a standard HS for all cases, as well as most of the approaches from the literature.

Keywords: Harmony search, location-allocation, inventory planning, vehicle routing, supply chain, return products.

1. Introduction

In a supply chain network design, the key element of the success both in profitability and productivity is its distribution network. The need for network design is to control the flows of the products and vehicles as well as the total cost. Nowadays, customers will return the products to the store for reasons such as mistakes while ordering the products, defect items, etc. Considering that the returned products usually re-enter the market channel after a simple refurbishing process, reverse logistics network in the green supply chain is gaining more attention among the industries.

Designing a distribution network involves three sub-problems: facility location problem, inventory control problem, and vehicle routing problem (Javid and Azad, 2010). These problems are highly related at the strategic, tactical and operational decision levels. Integrating them into one may provides an effective distribution network and relate much to the real-life problems. These problems are gaining considerable attention among the researchers to propose several solution methods with regards to the Location-Inventory-Routing Problem (LIRP). Different combinations of facility location decisions, inventory management decisions, and vehicle routing decisions give rise to different research problems in the forward and reverse logistics. In this paper, the LIRP simultaneously integrates the location decision of distribution centres, the inventory policies of opened depots, and the vehicle routing decision in serving customers, in which the returned products are re-furbished and then re-entered into the market. Vehicles that start and end at the same depot, distribute the products to satisfy the demand of customers and retrieve returned products through the forward and reverse flows respectively.

Many studies are concerned with the location-routing problem, but there are only a few studies that focus on the integration of LIRP. The LIRP under an e-supply chain has been solved by using Hybrid Genetic-Simulated Annealing (HGSA) algorithm and compared with the Genetic Algorithm (GA) (Li et al., 2013). They assumed that the returned products are in good conditions and can be re-sold to the customers. Liu et al. (2015) proposed a pseudo-parallel

GA integrating with SA. They solved a stochastic LIRP with non-defect returns and the proposed method outperformed the GA on the optimal solution. Then, Deng et al. (2016) solved the same problem as in Liu et al. (2015) but considered both defect and non-defect returns. A Hybrid Ant-Colony Optimisation (HACO) is compared with a standard ACO and the proposed method is efficient in solving the model. The forward and reverse logistics of LIRP are solved with a New Tabu Search (NTS) in Yuchi et al. (2016). In NTS, the second-best solution is accepted in the search process when the local optimum is reached.

Recently, many companies preferred to produce and distribute their own products. The Economic Production Quantity (EPQ) model is suitable to be practiced by the manufacturing companies as it can be seen nowadays the development of manufacturing companies are growing fast. Since most of the previous studies integrate the Economic Order Quantity (EOQ) model in the inventory, we adapted the mathematical model of the LIRP by replacing the EOQ with the EPQ model. The EPQ is more realistic in supply chain since it is demand dependent. The EOQ assumes that the demand remain steady throughout the year and does not account for economic fluctuation. By using the EPQ model, the inventory could be controlled by deciding the optimal production units based on the current usage of demand rate and production rate.

In this paper, we develop a LIRP model considering returned products and propose a Modified Harmony Search (MHS) algorithm to find the optimal solution to the problem. The proposed MHS algorithm is a harmony search with modification on local search neighbourhood techniques where we added additional two local search heuristic approaches which are 2-Opt and 3-Opt. The objectives are to minimise the total costs of operating depots, shipping products between the depots and the customers, as well as the inventory costs of the depots. The LIRP is the extension of the research from Misni and Lee (2019b) and Misni and Lee (2019a). Before the three sub-problems are integrated, the idea of solving the Vehicle Routing Problem (VRP) using HS algorithm is first proposed in Misni and Lee (2019b). Later, the problem is further extended into the Location-Routing Problem (LRP) in Misni and Lee (2019a). The remainders of the paper are organised as follows: the mathematical formulation of the LIRP is presented in Section 2, while the details of the proposed MHS framework for LIRP is described in Section 3. Computational results and discussion are given in Section 4. Lastly, conclusions and future research are addressed in Section 5.

2. Mathematical Formulation

The LIRP integrates three sub-problems: facility location problem, inventory control problem, and vehicle routing problem. These three different decision plannings: strategic, tactical, and operational, are solved sequentially. The LIRP involves several numbers of potential open depots and a set of customers to be served. The decision makings in this problem are:
1. to determine the number and location of optimal open depots,
2. to allocate the customers to the open depots,
3. to determine the vehicle routing between open depots and the customers,
4. to determine the optimal number of production quantity for each open depot.

In this paper, the objectives of LIRP are to minimise the total fixed operating cost of depots, the inventory cost including setup and holding cost, and the total distance travelled cost by the vehicles. The locations of possible depots and customers are given and the demands of each customer to be served by vehicles are predetermined. The maximum capacity limit for depots and vehicles are also provided. Figure 1 shows an example of LIRP in the supply chain with 3 depots and 9 customers. It can be seen from the Figure 1 that customers 1, 2, and 3 are served by depot 1, while the remaining 6 customers are served by depot 2 using two vehicles. Depot 3 is remain closed.

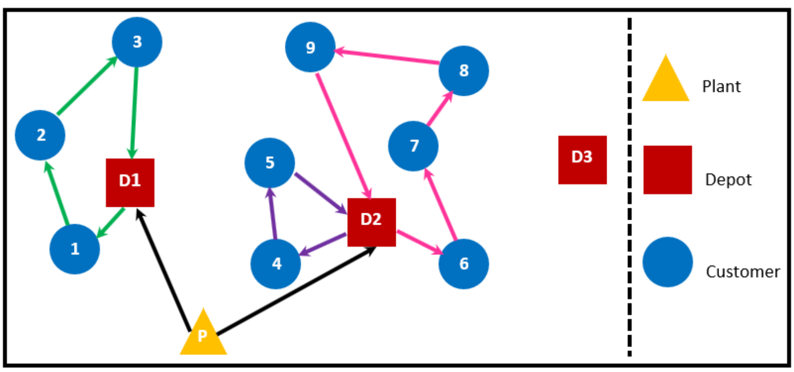

Figure 1: Example of a LIRP with 3 depots and 9 customers.

The mathematical modeling of LIRP is described as follows. The objectives and the constraints are given below:

Set:
I = sets of all depots $(i = 1, 2, \ldots, I)$
J = sets of all customers $(j = 1, 2, \ldots, J)$
K = sets of all vehicles $(k = 1, 2, \ldots, K)$

Input parameter:
D_j = demand of customer j
R_j = return of customer j
TU_{ijk} = total load on vehicle k of customer j at depot i
d_{ij} = distance from i to j
V_k = capacity of vehicle k
N = number of customer
F_i = fixed operating cost of depot i
DC = distance cost per miles
V_i = maximum throughput at depot i
Kc = setup cost per production
h = holding cost per unit inventory
P = production rate

Decision variables:
U_{lk} = auxiliary variable for sub-tour elimination constraints in vehicle k of customer l.
Q_i = optimal number of production quantity for each depot i.

$$z_i = \begin{cases} 1, & \text{if depot } i \text{ is open} \\ 0, & \text{otherwise.} \end{cases}$$

$$y_{ij} = \begin{cases} 1, & \text{if depot } i \text{ is assigned to customer } j \\ 0, & \text{otherwise.} \end{cases}$$

$$x_{ijk} = \begin{cases} 1, & \text{if arc } (i,j) \text{ is travelled by vehicle } k \\ 0, & \text{otherwise.} \end{cases}$$

$$\min \quad f = \sum_{i \in I} F_i z_i + \sum_{i \in I \cup J} \sum_{j \in I \cup J} \sum_{k \in K} DCd_{ij} x_{ijk} + \frac{1}{2} h \sum_{i \in I} Q_i \left(1 - \frac{\sum_{i \in I} \sum_{j \in J} (D_j + R_j) y_{ij}}{P} \right) + \frac{\sum_{i \in I} \sum_{j \in J} Kc(D_j - R_j) y_{ij}}{\sum_{i \in I} Q_i} \tag{1}$$

subject to:

$$\sum_{k \in K} \sum_{i \in I \cup J} x_{ijk} = 1, \forall j \in J \tag{2}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ijk} \leq 1, \forall k \in K \tag{3}$$

$$\sum_{i \in I \cup J} \sum_{j \in J} D_j x_{ijk} \leq V_k, \forall k \in K \tag{4}$$

$$\sum_{i \in I \cup J} \sum_{j \in J} R_j x_{ijk} \leq V_k, \forall k \in K \tag{5}$$

$$TUijk - (D_j - Rj) x_{ijk} \leq V_k, \forall i \in I, \forall j \in J, \forall k \in K \tag{6}$$

$$\sum_{i \in I \cup J} x_{ijk} - \sum_{j \in I \cup J} x_{jik} = 0, \forall i \in I, \forall k \in K \tag{7}$$

$$\sum_{j \in J} D_j y_{ij} \leq V_i z_i, \forall i \in I \tag{8}$$

$$U_{lk} - U_{jk} + N x_{ljk} \leq N - 1, \forall l, j \in J, \forall k \in K \tag{9}$$

$$\sum_{u \in I \cup J} x_{iuk} + x_{ujk} - y_{ij} \leq 1, \forall i \in I, \forall j \in J, \forall k \in K \tag{10}$$

$$y_{ij} \in \{0, 1\}, \forall i \in I, \forall j \in J \tag{11}$$

$$x_{ijk} \in \{0, 1\}, \forall i \in I \cup J, \forall j \in I \cup J, \forall k \in K \tag{12}$$

$$z_i \in \{0, 1\}, \forall i \in I \tag{13}$$

$$U_{lk} \geq 0, \forall l \in J, \forall k \in K \tag{14}$$

$$Q_i \geq 0, \forall i \in I \tag{15}$$

The objectives function of LIRP (Eqn.(1)) are to minimise the total fixed operating cost of depots, the total distance travelled cost by the vehicles, and the total cost of setup production and holding inventory. Constraints in Eqn.(2)

and (3) indicate that each of the customers has to be assigned in a single route and it can be served by only one vehicle. The total demands and total returns at each route cannot exceed the vehicle capacity limit and the total load on the vehicle at any arc must not exceed the vehicle capacity. These constraints are shown in Eqn.(4)−(6), respectively. Eqn.(7) states that vehicle must start and end at the same depot. Besides the vehicle capacity limit, the capacity constraint for the depot is given in Eqn.(8). Eqn.(9) represents the new sub tour elimination constraint and Eqn.(10) specifies that the customer will be assigned to the depot if there is a route from that depot. The binary values on the decision variables and the positive values for the auxiliary variable are defined in Eqn.(11)−(15), respectively.

The formulation of EPQ and EOQ model is different in the objective function at the holding inventory cost. The cost of holding the inventory in EPQ model depends on the value of production rate and the usage of products are occurs continuously. Inventory increases at rate $P - D$ units per unit time until the production is completed. Hence the formulation of inventory for EPQ model is given as follows:

$$I = \frac{Q}{P}(P - D), \tag{16}$$

To solve the LIRP problem, the location and allocation of the depots and customers are solved first. In the location-allocation problem, each of the customers is initially assigned to the nearest depot based on the Euclidean distance formulation. The customers will be moved around to the possible depots during the process of improvisation. The vehicle capacity limit constraint has been relaxed to minimise the number of open depots and assumed only one vehicle is being used at each depot. However, the depot capacity limit should not be violated during the process.

After the number and location of depots are determined, the allocation of customers at each depot needs to be sequenced and divided into vehicles according to the vehicle capacity limit. This process is performed among the open depots only. The process of local optimisation is focusing within the open depots. Both depot capacity limits and vehicle capacity limits should not be violated. Then lastly, the inventory part is included during the process of minimising the cost.

3. Modified Harmony Search Algorithm

Harmony Search (HS) is a population-based metaheuristic algorithm that mimics the music improvisation of a group orchestra. The musician will improvise their harmony with these three options: (1) select any pitch that has been played from the previous harmony memory, (2) fix the pitch that sounds similar to any previous pitch in memory, or (3) compose a new music harmonisation. The process of obtaining the optimal solutions is similar to this efficient search for a perfect state of harmony (Geem et al., 2001).

Over the years, there are many modifications on the HS by the researchers, especially when dealing with vehicle routing and facility location problems. In this study, the steps and the modifications proposed for the MHS algorithm are detailed below.

Step 1: Initialisation
In HS, several solutions are generated to produce an initial population. The solutions can be generated either randomly or by the heuristics. In the proposed MHS, a random initial population called Harmony Memory (HM) is created and sorted according to their fitness values. The number of solutions in the HM is the Harmony Memory Size (HMS). A HM can be described in the following matrix:

$$HM = \left[\begin{array}{cccc|c} x_1^1 & x_2^1 & \cdots & x_n^1 & f(x^1) \\ x_1^2 & x_2^2 & \cdots & x_n^2 & f(x^2) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_1^{HMS} & x_2^{HMS} & \cdots & x_n^{HMS} & f(x^{HMS}) \end{array}\right], \tag{17}$$

where,
x_i^j = decision variable for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots,$HMS,
$f(x^j)$ = fitness function of x^j for $j = 1, 2, \ldots,$HMS.

Step 2: Parameter Setting
The parameters setting used in the proposed MHS algorithm are: Harmony Memory Considering Rate (HMCR), Pitch Adjusting Rate (PAR), HMS, and the stopping criterion. An appropriate value of HMCR will lead to the choices of good solutions as an element of new solutions. The convergence will be slow if the value of HMCR is too high. Some potential good solutions are not well explored. But if it is too low, only a few good solutions are selected for the

next iteration. Therefore, to use the memory effectively, the value of HMCR should be between 0.7 and 0.95 (Yang, 2009).

Bandwidth (bw) determines the amount of changes that may occur in the pitch adjustment. The use of bw is dependent on the variable either it is discrete or continuous. The narrow value of bw will limit the solutions to be explored to a single portion of the search space. The speed of convergence in a standard HS algorithm may be slow if the value of PAR is too small. Besides, a high value of PAR is also not recommended. It can cause solutions to be stuck around a few potential optimal and easily get trapped in a local optima. Due to this, Yang (2009) stated that the value of PAR is suggested to be between 0.1 and 0.5 and the lower and upper bound of bw are generally set as 1% and 10% respectively. However, this range may be suitable and valid for some problems only.

To balance the exploration and exploitation, the proposed MHS uses a dynamic value of HMCR and PAR by using the following formulation:

$$HMCR_{it} = HMCR_{max} - (HMCR_{max} - HMCR_{min})\frac{it}{MaxIt}, \tag{18}$$

$$PAR_{it} = PAR_{max} - (PAR_{max} - PAR_{min})\frac{it}{MaxIt}, \tag{19}$$

where,
it = the current iteration,
$MaxIt$ = maximum iterations,
$HMCR_{max}$ = maximum value of the HMCR,
$HMCR_{min}$ = minimum value of the HMCR,
PAR_{max} = maximum value of the PAR,
PAR_{min} = minimum value of the PAR.

This formulation is introduced by Mahdavi et al. (2007), but with a slight modifications. The proposed HMCR and PAR are reduced gradually when there is no improvement found in the solution. In this problem, the range values of HMCR and PAR are set to be [0.7,0.95] and [0.3,0.9] respectively. The reason for reducing the HMCR slowly is to increase the probability of exploring more solution space, not in the HM. Hence, the global optimal can be attained (Moh'd Ali and Mandava, 2011). The dynamic value of HMCR and PAR can avoid the solutions getting trapped in the local optimum quickly.

Step 3 : Improvisation Process

Generate the New Solution

A standard HS utilise a single search solution to evolve. This makes the algorithm converged slowly to the optimal solution. To increase the speed of convergence, several new solutions are generated at each iteration and called as HM_{new} in the proposed MHS. The size of HM_{new} is smaller than the size of HM. A new solution is taken from the HM randomly with the probability of HMCR, otherwise, it will be generated randomly within the range of the harmony vector.

Local Neighbourhood Search

To increase the intensification of the method, the proposed MHS implements the multi-local neighbourhood search. The techniques of local search are divided into three: within a depot, between two depots and within the vehicle route. Five techniques of neighbourhood search are proposed: swap (within depot or between depots), insertion, relocation, 2-Opt, and 3-Opt. The local searches are applied to the new solution with the probability of PAR. At each iteration, the process of generating a new solution and local search is repeated until the number of harmony vectors equal to the size of HM_{new}. The choice of local neighbourhood search is randomly selected.

Swap: exchange the position of two random customers within the same route or between different routes. Swapping can be done within a depot or between the depots.
Insertion: insert a customer in between two other random customers within the same depot.
Relocation: relocate the customers from the current depot to a new depot.
2-Opt: swapping two customers in the same vehicle route and reverse the sub-string between the swapped customers.
3-Opt: deleting three edges of the customers in the same vehicle route and create another three sub-tours in three possible ways: non-reversing sub-string, with one reversing sub-string or with two reversing sub-string.

Swapping within a depot and insertion are two techniques that involve the searching of the neighbourhood within the same depot while swapping between two depots and relocation are techniques that require two different depots. The 2-Opt and 3-Opt are techniques that involve the movement among the vehicle routes in the same depot. In the location-allocation phase and multi-depot vehicle routing phase, the empty depots will not be involved during the local neighbourhood search processes. The example of these techniques are graphically shown in Figure 2 – 7.

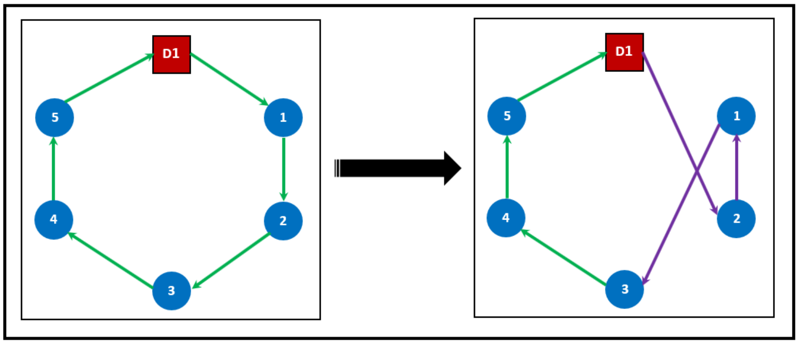

Figure 2: Swap within depot: customer 1 and 2 are swapped within the same depot.

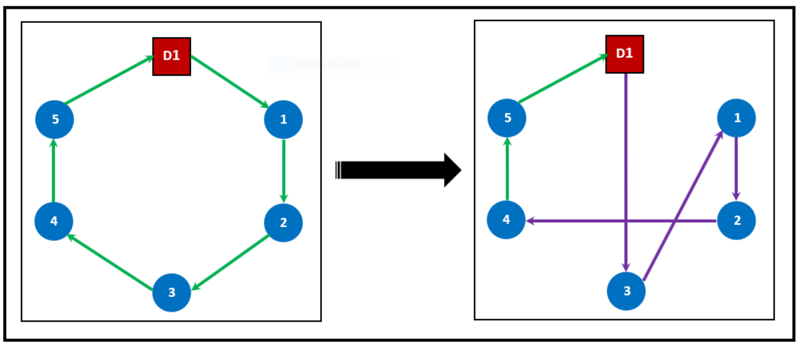

Figure 3: Insertion: customer 3 are inserted in between depot and customer 1 in the route sequence.

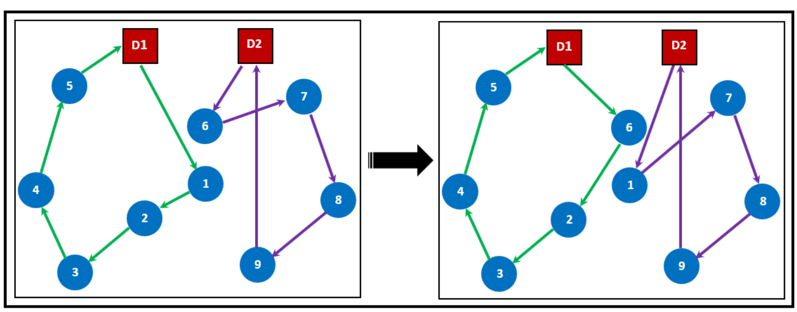

Figure 4: Swap between depots: customer 6 in depot 1 is swapped with customer 1 in depot 2.

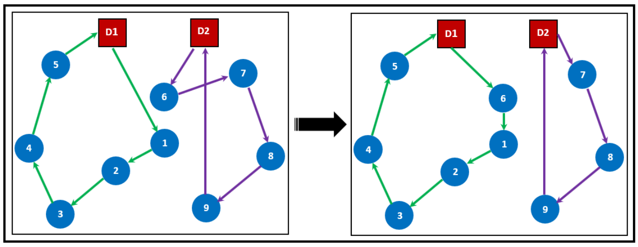

Figure 5: Relocation: customer 6 in depot 2 is moved to depot 1.

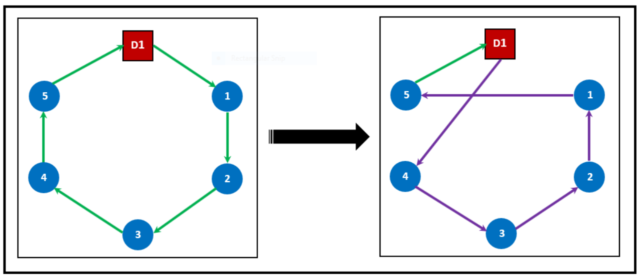

Figure 6: 2-Opt: swap customer 1 and 4 in the same route and the route sequence between swapped customers are reversed.

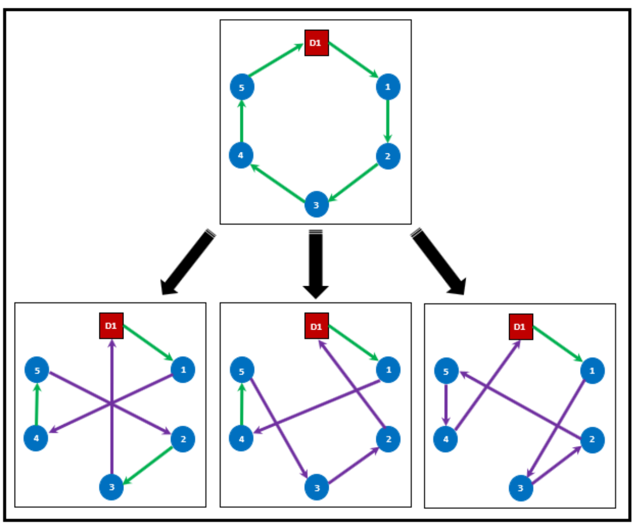

Figure 7: 3-Opt: three possibilities of 3-Opt which are either non-reversing, one reversing or two reversing substring.

Step 4: Update HM
To update the solutions in HM, the HM_{new} created at each iteration is combined with the current HM. The solutions are sorted according to the value of the objective function. The best solutions with the size of HMS will be kept for the next iteration.

Step 5: Stopping Criteria
The proposed MHS algorithm will be terminated when 100 consecutive non-improving iterations are performed.

The algorithm of the proposed MHS is given below.

```
Algorithm HS
begin
Initialisation phase:
Define harmony vectors and fitness function
Define parameter setting: HMCR_max, HMCR_min,PAR_max,PAR_min, HMS, MaxIt
Generate initial population of HM
Improvisation phase:
Do the location-allocation
while (stopping criteria is not met) do
    while (no. of harmony vectors < size of HM_new) do
        Calculate HMCR_it and PAR_it
        if (rand ≤ HMCR_it) then
            choose solution from the HM randomly
            if (rand ≤ PAR_it and new solution ∈ HM_best) then
                perform the route-allocation and local neighborhood search
            else if (rand ≤ PAR_it and new solution ∈ HM_worst) then
                perform the route-allocation
            else
                keep the harmony vectors
            end if
        else
            explore the other harmony vectors space
        end if
        Do the inventory planning
        Calculate the new fitness function of each harmony vector
    end while
    Combine the HM with HM_new. Sort the fitness function
    Update the best harmony vector with size of HMS
end while
Return the best solution vectors of harmony = first solution in HM
end
```

4. Computational Results and Discussion

The proposed MHS is implemented in MATLAB software R2017b and tested with three benchmark dataset of Perl, Gaskell, and Christofides with additional generated data of returned products and production rate. The characteristics of the dataset are given in Table 1.

Table 1: Perl, Gaskell and Christofides dataset

Instances	Customer	Depot	Depot Capacity	Vehicle Capacity
Perl 1	12	2	280	140
Perl 2	55	15	550	120
Perl 3	85	7	850	160
Gaskell 1	21	5	15000	6000
Gaskell 2	22	5	15000	4500
Gaskell 3	29	5	15000	4500
Gaskell 4	32	5	35000	8000
Gaskell 5	36	5	15000	250
Christofides 1	50	5	10000	160
Christofides 2	75	10	10000	140
Christofides 3	100	10	10000	200

Since there is no study on the LIRP with the EPQ model from the literature, to assess the performance of the proposed MHS, the LRP will be solved first. The proposed MHS is compared with other heuristics and metaheuristics found in the literature. The comparative results of the proposed MHS with heuristic method (Heuristic) (Perl and Daskin, 1985), Simulated Annealing in Wu et al. (2002) (SA-W) and in Zhang et al. (2015) (SA-Z), Hybrid Tabu Search and Ant Colony (Tabu-ACO) (Wang et al., 2005) and Particle Swarm Optimisation (PSO) (Marinakis and Marinaki, 2008) are provided in Table 2 for the Perl dataset. While for Gaskell and Christofides dataset, the proposed MHS is compared with different variants of particle swarm which are standard PSO, PSO with MPNS-GRASP (PSO-MPNS-GRASP), PSO with MPNS-GRASP and ENS (PSO-MPNS-GRASP-ENS) and hybrid PSO (HybPSO-LRP) in Marinakis and Marinaki (2008). The proposed algorithm is also compared with the Honey Bee Mating Optimisation (HBMO) in Marinakis et al. (2008). The results of these comparisons are presented in Table 3 and Table 4, respectively. A Standard HS (SHS) and Proposed HS (PHS) from Misni and Lee (2019a) are also included for all experiments.

Table 2: Comparative Results of Perl Datase for LRP

	Perl 1	Perl 2	Perl 3
Heuristic	**203.97**	1146.93	1657.60
SA-W	**203.97**	1119.83	1656.72
Tabu-ACO	**203.97**	1139.87	1642.57
PSO	**203.97**	1135.90	1656.90
SA-Z	**203.97**	1115.40	**1642.00**
SHS	**203.97**	1158.80	1832.10
PHS	**203.97**	1113.24	1652.99
Proposed MHS	**203.97**	**1113.20**	1652.18

Table 3: Comparative Results of Gaskell Dataset for LRP

	Gaskell 1	Gaskell 2	Gaskell 3	Gaskell 4	Gaskell 5
PSO	437.1	592.1	512.1	574.1	470.7
PSO-MPNS-GRASP	435.9	591.8	512.1	571.7	470.7
PSO-MPNS-GRASP-ENS	435.9	591.7	512.1	571.7	470.7
HybPSO-LRP	432.9	588.5	512.1	570.8	470.7
HBMO	431.9	587.2	512.1	569.8	470.7
SHS	443.7	598.0	541.2	576.2	520.7
PHS	431.7	577.9	511.8	559.9	464.1
Proposed MHS	**429.8**	**575.2**	**497.9**	**557.7**	**464.1**

Table 4: Comparative Results of Christofides Dataset for LRP

	Christofides 1	Christofides 2	Christofides 3
PSO	582.7	888.9	895.7
PSO-MPNS-GRASP	582.7	887.1	893.2
PSO-MPNS-GRASP-ENS	582.7	886.9	891.5
HybPSO-LRP	582.7	886.3	889.4
HBMO	582.7	886.3	889.4
SHS	602.9	953.3	977.6
PHS	587.7	886.7	891.6
Proposed MHS	**575.4**	**885.0**	**887.8**

As compared to the SHS and PHS, the proposed MHS performed significantly better for all instances in Gaskell and Christofides, as well as in Perl 2 and Perl 3. All algorithms managed to find the optimal solution in Perl 1 which is 203.97 since the number of customers and depots are small. The SHS did not perform well for medium and large size. Based on the results, the proposed MHS obtained the best solution for Perl 1 and Perl 2. However, for Perl 3, results in Wang et al. (2005) and Zhang et al. (2015) are slightly better than the proposed MHS. The proposed MHS outperformed the solutions in Misni and Lee (2019a) for all instances of Perl, Gaskell and Christofides. This shows that the proposed MHS are better than PHS in Misni and Lee (2019a) and hence it will be used for solving the LIRP with EPQ model.

For LIRP with EPQ model, the same benchmark dataset are used but with additional generated data for returned products and production rate. The data were generated randomly by using a uniform distribution. The production rate is set to be greater than the demand rate $p > \sum d_j$ and the returned products are generated by the uniform distribution of $r_j \sim U(0, d_j)$. These problems are then solved by the proposed MHS and compared with the SHS only. Note that, the EOQ model determines the optimal number of products to be ordered while the EPQ model determines for the optimal number of production quantity per batch. Hence, the general formulation of the Q for EPQ model is different with EOQ model and is given as follows:

$$Q = \sqrt{\frac{2KD}{h}}\sqrt{\frac{P}{P-D}}, \tag{20}$$

where,
Q = optimal number of production quantity,
K = setup cost per production,
D = total demand,
h = holding cost per unit inventory,
P = production rate,

The inventory in the EPQ relies on the demand and return of the allocated customers at each depot. The cost of the LIRP could be minimised when the cost of facilities, inventory and distance cost are reduced. The results for all instances are shown in Table 5 and since there is no benchmark for the EPQ model in LIRP, the comparison is done between the proposed MHS and SHS only. It can be seen that the proposed MHS outperforms the solutions in SHS for all dataset of Perl, Gaskell and Christofides. The LIRP gives the lower cost when the LRP is minimised. The inventory cost in LIRP can be minimised by finding the optimal number of production quantity by using the formulation.

Table 5: Comparative Results of Perl, Gaskell and Christofides Dataset for LIRP with EPQ

	SHS	Proposed MHS
Perl 1	**284.22**	**284.22**
Perl 2	1408.64	**1363.08**
Perl 3	2112.46	**1933.35**
Gaskell 1	1590.62	**1585.72**
Gaskell 2	1020.36	**997.56**
Gaskell 3	814.37	**771.07**
Gaskell 4	1017.60	**999.10**
Gaskell 5	592.48	**535.08**
Christofides 1	784.73	**757.13**
Christofides 2	1122.43	**1052.78**
Christofides 3	1299.19	**1209.38**

Figure 8 below illustrates the graphical layout of the customers' route delivery assigned at each open depot in Gaskell 1 problem instance. As can be seen here, only two depots will be opened with the facility cost of 50 each. The customers' route at depot 1 with inventory cost of 580.22 are D1 → C21→ C19→ C16→ D1 with cost of 58.38 and D1 → C17→ C20→ C18→ C15→ C12→ C14→ D1 with cost of 92.10. For depot 2, the cost of inventory is 575.70 and the customers' route are D2 → C9→ C7→ C5→ C2→ C1→ C6 → C8 → D2 and D2 → C3→ C4→ C11→ C13→ C10 → D2 with cost of 83.80 and 95.50 respectively. Both depots required two vehicles to deliver the customers' demand.

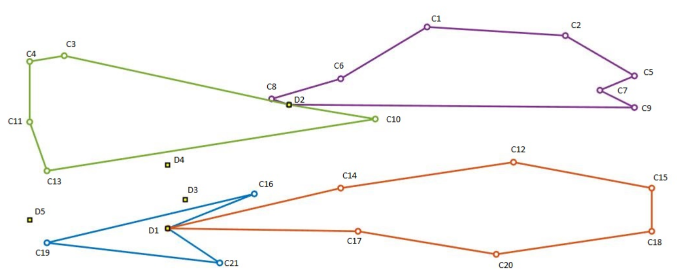

Figure 8: Layout of Gaskell 1 by MHS for LIRP.

5. Conclusions and Future Research

The integration of facility location, inventory planning, and vehicle routing is one of the most complicated problems in the supply chain network design. Since the problem is NP-hard, a population-based metaheuristic, Harmony Search is proposed. The reverse logistics of the model that considers the returned products from the customers is included during the process of optimisation. Computational experiments on three sets of benchmark instances show that the proposed MHS with the dynamic value of HMCR and PAR and the multi-local neighbourhood search techniques is successful in finding better solutions as compared to a standard HS, as well as other metaheuristics from the literature. For future direction, the LIRP can be solved and compared with other metaheuristic methods such as Differential Evolution, Genetic Algorithm, Tabu Search, and Simulated Annealing. A hybrid between metaheuristic approaches could also enhanced the performance of the algorithm. The environmental issue such as the CO_2 emission can also be integrated into the model to be solved.

Acknowledgment

This research was supported by the Geran Putra-Inisiatif Putra Siswazah (GP-IPS/2017/9579400) funded by Universiti Putra Malaysia (UPM).

References

Deng, S., Li, Y., Guo, H., and Liu, B. (2016). Solving a closed-loop location-inventory-routing problem with mixed quality defects returns in e-commerce by hybrid ant colony optimization algorithm. *Discrete Dynamics in Nature and Society*, 2016. https://doi.org/10.1155/2016/6467812.

Geem, Z. W., Kim, J. H., and Loganathan, G. V. (2001). A new heuristic optimization algorithm: Harmony search. *Simulation*, 76(2):60–68.

Javid, A. A. and Azad, N. (2010). Incorporating location, routing and inventory decisions in supply chain network design. *Transportation Research Part E: Logistics and Transportation Review*, 46(5):582–597.

Li, Y., Guo, H., Wang, L., and Fu, J. (2013). A hybrid genetic-simulated annealing algorithm for the location-inventory-routing problem considering returns under e-supply chain environment. *The Scientific World Journal*, 2013. https://doi.org/10.1155/2013/125893.

Liu, B., Chen, H., Li, Y., and Liu, X. (2015). A pseudo-parallel genetic algorithm integrating simulated annealing for stochastic location-inventory-routing problem with consideration of returns in e-commerce. *Discrete Dynamics in Nature and Society*, 2015. https://doi.org/10.1155/2015/586581.

Mahdavi, M., Fesanghary, M., and Damangir, E. (2007). An improved harmony search algorithm for solving optimization problems. *Applied Mathematics and Computation*, 188(2):1567–1579.

Marinakis, Y. and Marinaki, M. (2008). A particle swarm optimization algorithm with path relinking for the location routing problem. *Journal of Mathematical Modelling and Algorithms*, 7:59–78. https://doi.org/10.1007/s10852-007-9073-6.

Marinakis, Y., Marinaki, M., and Matsatsinis, N. (2008). Honey bees mating optimization for the location routing problem. *IEEE International Engineering Management Conference*, pages 1–5. https://doi.org/10.1109/IEMCE.2008.4618013.

Misni, F. and Lee, L. S. (2019a). Harmony search algorithm for location-routing problem in supply chain network design. *ASM Science Journal*, 12(5):48–56.

Misni, F. and Lee, L. S. (2019b). Harmony search for multi-depot vehicle routing problem. *Malaysian Journal of Mathematical Sciences*, 13(3):311–328.

Moh'd Ali, O. and Mandava, R. (2011). The variants of the harmony search algorithm: An overview. *Artificial Intelligence Review*, 36(1):49–68.

Perl, J. and Daskin, M. S. (1985). A warehouse location-routing problem. *Transportation Research Part B: Methodological*, 19(5):381–396.

Wang, X., Sun, X., and Fang, Y. (2005). A two-phase hybrid heuristic search approach to the location-routing problem. *IEEE International Conference on Systems, Man and Cybernetics*, 4:3338–3343.

Wu, T. H., Low, C., and Bai, J. W. (2002). Heuristic solutions to multi-depot location-routing problems. *Computers & Operations Research*, 29(10):1393–1415.

Yang, X. S. (2009). *Harmony Search as a Metaheuristic Algorithm. In Music-inspired Harmony Search Algorithm.* Berlin, Hiedelberg: Springer.

Yuchi, Q., He, Z., Yang, Z., and Wang, N. (2016). A location-inventory-routing problem in forward and reverse logistics network design. *Discrete Dynamics in Nature and Society*, 2016. https://doi.org/10.1155/2016/3475369.

Zhang, Y., Qi, M., Lin, W. H., and Miao, L. (2015). A metaheuristic approach to the reliable location routing problem under disruptions. *Transportation Research Part E: Logistics and Transportation Review*, 83:90–110.