Genetic Algorithm for Solving Location Problem in a Supply Chain Network with Inbound and Outbound Product Flows

Suprayogi[1]*, Senator Nur Bahagia[1], Yudi Thaddeus[1]

Abstract: This paper considers a location problem in a supply chain network. The problem addressed in this paper is motivated by an initiative to develop an efficient supply chain network for supporting the agricultural activities. The supply chain network consists of regions, warehouses, distribution centers, plants, and markets. The products include a set of inbound products and a set of outbound products. In this paper, definitions of the inbound and outbound products are seen from the region's point of view. The inbound product is the product demanded by regions and produced by plants which flows on a sequence of the following entities: plants, distribution centers, warehouses, and regions. The outbound product is the product demanded by markets and produced by regions and it flows on a sequence of the following entities: regions, warehouses, and markets. The problem deals with determining locations of the warehouses and the distribution centers to be opened and shipment quantities associated with all links on the network that minimizes the total cost. The problem can be considered as a strategic supply chain network problem. A solution approach based on genetic algorithm (GA) is proposed. The proposed GA is examined using hypothetical instances and its results are compared to the solution obtained by solving the mixed integer linear programming (MILP) model. The comparison shows that there is a small gap (0.23%, on average) between the proposed GA and MILP model in terms of the total cost. The proposed GA consistently provides solutions with least total cost. In terms of total cost, based on the experiment, it is demonstrated that coefficients of variation are closed to 0.

Keywords: Location problem, supply chain network, mixed integer linear programming model, genetic algorithm.

Introduction

This paper considers a location problem in a supply chain network. The problem addressed in this paper has been discussed by Suprayogi *et al.* [1] where it is motivated by an initiative to develop an efficient supply chain network for supporting the agricultural activities. The supply chain network can help farmers to get agricultural activity supporting products (e.g. fertilizers, pesticides, and seeds) at lower cost and better supply. The supply chain network also can support farmers for distributing their agricultural products (e.g. rice) to markets. The supply chain network is designed to establish distribution centers and warehouses that can act as intermediaries.

The supply chain network consists of regions, warehouses, distribution centers, plants, and markets. The products include a set of inbound products and a set of outbound products. In this paper, definitions of the inbound and outbound products are seen from the region's point of view.

The inbound product is the product demanded by regions and produced by plants which flows on a sequence of the following entities: plants, distribution centers, warehouses, and regions. The outbound product is the product demanded by markets and produced by regions and it flows on a sequence of the following entities: regions, warehouses, and markets. The supply chain network considered in this paper is illustrated in Figure 1.

The problem deals with determining locations of the warehouses and the distribution centers to be opened and shipment quantities associated with all links on the network that minimizes the total cost.
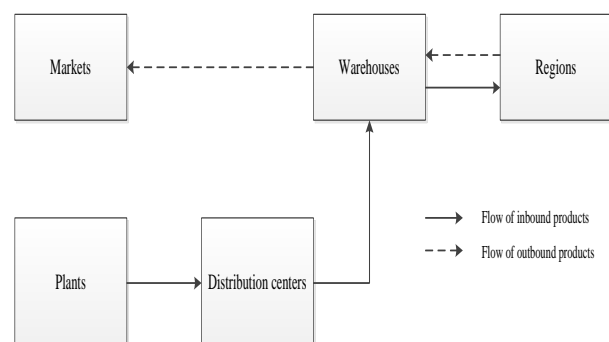

Figure 1. Illustration of the supply chain network

[1] Faculty of Technology, Industrial Engineering Department, Institut Teknologi Bandung, Jl. Ganesha 10, Bandung 40132, Indonesia. Email: yogi@mail.ti.itb.ac.id, senatornurb@yahoo.co.id, tuing2ing@gmail.com.

* Corresponding author

A mixed integer linear programing (MILP) model to represent the problem has been proposed by Suprayogi *et al.* [1]. They gave an illustrative example and solution of the problem.

The problem can be considered as a strategic supply chain network problem. Min and Zhou [2] has provided a review and taxonomies in supply chain modeling. In addition, the problem addressed in this paper integrates the facility location decisions with other relevant decisions in a supply chain network. Melo *et al.* [3] have pointed out the important roles of facility decision problems in the strategic supply chain network. They gave a literature review of facility location models in the context of supply chain management. According to the supply chain structure mentioned by Melo *et al.* [3], this problem has the following features: multiple location layers, multiple commodities, single period, and deterministic.

The supply chain network discussed in this paper involves simultaneously forward and reverse flows. In literature, this typical supply chain network can be called as the supply chain network with bidirectional flows as mentioned in De Rosa *et al.* [4] and Zhang *et al.* [5].

The facility location problem is one of hard combinatorial optimization problems. Solving the MILP model can be applied only for small problems. In this paper, a solution approach based on genetic algorithm (GA) is proposed. GA is one of metaheuristic approaches where it is inspired from the natural selection process. The proposed GA in this paper is adapted from Costa *et al.* [6] where they proposed a new encoding/decoding procedure for designing a single product three-stage supply chain network. Using this procedure, infeasible distribution flows can be avoided in the decoding process.

Methods

Problem Definition

The problem definition is taken from Suprayogi *et al.* [1]. There are five entities in the supply chain network, i.e.: plants, distribution centers, warehouses, regions, and market. There are a set of inbound products produced by plants. The supply capacities of inbound products are known. There are a set of regions where each region requires the inbound products. The inbounds product is sent from plant to regions through distribution centers and warehouses. There is a set of outbound products demanded by markets and produced by regions. The outbound products are delivered to markets through warehouses.

There is a set of location candidates of warehouses and a set of location candidates of distribution centers. The problem deals with determining locations of warehouses and distribution centers to be opened and shipment quantities to minimize the total cost. The total cost consists of costs for opening warehouses and distribution centers and costs of transporting both inbound and outbound products.

Characteristics of the supply chain network to be designed must meet the following policies: (a) A distribution center can receive an inbound product from one plant or more. (b) A warehouse must receive all inbound products from a particular distribution center. (c) A region must receive all inbound products from a particular warehouse. (d) All outbound products from a region must be delivered to a warehouse. (e) The warehouse delivering inbound products to a region and receiving outbound products of a region is same. (f) Each market can receive outbound products from several warehouses.

Mixed Integer Linear Programing Model

Notations used in the MILP model are as follows:
Sets:

I	region set
J	warehouse set
K	distribution center set
S	plant set
M	market set
A	inbound product set
B	outbound product set

Indices:

i	region index
j	warehouse index
k	distribution center index
s	plant index
m	market index
a	inbound product index
b	outbound product index

Parameters:

DI_{ia}	demand of region i for inbound product a
SO_{ib}	supply of region i for outbound product b
FCX_k	fixed cost of establishing distribution center k
FCX_j	fixed cost of establishing warehouse j
$CAPIX_k$	inbound capacity of distribution center k
$CAPIY_j$	inbound capacity of warehouse j
$CAPOY_j$	outbound capacity of warehouse j
$CAPIS_{sa}$	inbound capacity of plant s for product a
$CAPOM_{mb}$	outbound capacity of market m for product b
TSK_{ska}	unit transportation cost of inbound product a from plant s to distribution center k

TKJ_{kja} — unit transportation cost of inbound product a from distribution center k to warehouse j

TJI_{jia} — unit transportation cost of inbound product a from warehouse j to region i

TIJ_{ijb} — unit transportation cost of outbound product b from region i to warehouse j

TJM_{jmb} — unit transportation cost of outbound product b from warehouse j to market m

Ω — very large positive number

Decision variables:

QSK_{ska} — shipment quantity of inbound product a from plant s to distribution center k

QKJ_{kja} — shipment quantity of inbound product a from distribution center k to warehouse j

QJI_{jia} — shipment quantity of inbound product a from warehouse j to region i

QIJ_{ijb} — shipment quantity of outbound product b from region i to warehouse j

QJM_{jmb} — shipment quantity of outbound product b from warehouse j to market m

X_k — binary variable representing the decision for opening distribution center k ($X_k = 1$ if distribution center k is opened, $X_k = 0$ if otherwise)`

Y_j — binary variable representing the decision for opening warehouse j ($Y_j = 1$ if warehouse j is opened, $Y_j = 0$ if otherwise)

ZIJ_{ij} — binary variable representing the decision for shipping products between region i and warehouse j ($ZIJ_{ij} = 1$ if products are shipped between region i and warehouse, $ZIJ_{ij} = 0$ if otherwise)

ZKJ_{kj} — binary variable representing the decision for shipping products between distribution center k and warehouse j ($ZKJ_{kj} = 1$ if products are shipped distribution center k and warehouse j, $ZKJ_{kj} = 0$ if otherwise)

Performance measure:

TC — total cost

According to Suprayogi et al. [1], the problem is formulated into a MILP model as follows:

Minimize
$TC =$
$\sum_{k \in K} FCX_k \cdot X_k + \sum_{j \in J} FCY_j \cdot Y_j +$
$\sum_{s \in S} \sum_{k \in K} \sum_{a \in A} TSK_{ska} \cdot QSK_{ska} +$
$\sum_{k \in K} \sum_{j \in J} \sum_{a \in A} TKJ_{kja} \cdot QKJ_{kja} +$
$\sum_{j \in J} \sum_{i \in I} \sum_{a \in A} TJI_{jia} \cdot QJI_{jia} + \sum_{i \in I} \sum_{j \in J} \sum_{b \in B} TIJ_{ijb} \cdot$
$QIJ_{ijb} + \sum_{j \in J} \sum_{m \in M} \sum_{b \in B} TJM_{jmb} \cdot QJM_{jmb}$ (1)

subject to:
$$ZIJ_{ij} \leq Y_j; \forall i \in I, \forall j \in J \tag{2}$$

$$ZKJ_{kj} \leq X_k; \forall k \in K, \forall j \in J \tag{3}$$
$$\sum_{j \in J} ZKJ_{kj} = 1; \forall i \in I \tag{4}$$
$$\sum_{j \in J} ZKJ_{jk} = Y_j; \forall j \in J \tag{5}$$
$$QJI_{jia} \leq \Omega \cdot ZIJ_{ij}; \forall j \in J, \forall i \in I, \forall a \in A \tag{6}$$
$$QKJ_{kja} \leq \Omega \cdot ZKJ_{kj}; \forall k \in K, \forall j \in J, \forall a \in A \tag{7}$$
$$QIJ_{ijb} \leq \Omega \cdot ZIJ_{ij}; \forall i \in I, \forall j \in J, \forall a \in A \tag{8}$$
$$\sum_{i \in I} \sum_{a \in A} QJI_{jia} \leq CAPIY_j; \forall j \in J \tag{9}$$
$$\sum_{j \in J} \sum_{a \in A} QKJ_{kja} \leq CAPIX_k; \forall k \in K \tag{10}$$
$$\sum_{k \in K} QSK_{ska} \leq CAPIS_{sa}; \forall s \in S, \forall a \in A \tag{11}$$
$$\sum_{i \in I} \sum_{b \in B} QIJ_{ijb} \leq CAPOY_j; \forall j \in J \tag{12}$$
$$\sum_{j \in J} QJM_{jmb} \leq CAPOM_{mb}; \forall m, \forall b \tag{13}$$
$$\sum_{j \in J} QJI_{jia} = DI_{ia}; \forall i, \forall a \tag{14}$$
$$\sum_{k \in K} QKJ_{kja} = \sum_i QJI_{jia}; \forall j, \forall a \tag{15}$$
$$\sum_{s \in S} QSK_{ska} = \sum_j QKJ_{kja}; \forall k, \forall a \tag{16}$$
$$\sum_{j \in J} QIJ_{ijb} = SO_{ib}; \forall i, \forall b \tag{17}$$
$$\sum_{m \in M} QJM_{jmb} = \sum_{i \in I} QIJ_{ijb}; \forall j, \forall b \tag{18}$$
$$QSK_{ska} \geq 0; \forall s \in S, \forall k \in k, \forall a \in A \tag{19}$$
$$QKJ_{kja} \geq 0; \forall k \in K, \forall j \in J, \forall a \tag{20}$$
$$QJI_{jia} \geq 0; \forall j \in J, \forall i \in I, \forall a \in A \tag{21}$$
$$QIJ_{ijb} \geq 0; \forall i \in I, \forall j \in J, \forall b \in B \tag{22}$$
$$QJM_{jmb} \geq 0; \forall j \in J, \forall m \in M, \forall b \in B \tag{23}$$
$$X_k \in \{0, 1\}; \forall k \in K \tag{24}$$
$$Y_j \in \{0, 1\}; \forall j \in J \tag{25}$$
$$ZIJ_{ij} \in \{0, 1\}; \forall i \in I, \forall j \in J \tag{26}$$
$$ZKJ_{kj} \in \{0, 1\}; \forall k \in K, \forall j \in J \tag{27}$$

The objective function to be minimized is formulated in Equation (1). This equation represents the total cost including costs for opening distribution centers and warehouses and costs of transporting products. Constraints (2) and (3) ensure that product flows occur if entities (warehouses and distribution) are opened. Constraints (4) ensure that a region must be linked by a warehouse. Constraints (5) guarantee that a warehouse is linked by a distribution center. Constraints (6)-(8) are linking constraints. Capacity constraints of all entities are formulated in Constraints (9)-(13). Constraints (14)-(18) act as flow conservation constraints. Constraints (19)-(27) represent the decision variables constraints.

Genetic Algorithm

Outline of the Proposed GA

The GA approach starts by generating an initial population consisting a set of chromosomes. To represent the solution, a decoding procedure is carried out. Based on the solution, the total cost is calculated and it represent the fitness values. Three GA operators are applied, i.e., elitism, crossover, and mutation, to produce a new population. This process is repeated until the maximum generation is achieved. Figure 2 shows the flowchart of the proposed GA approach.
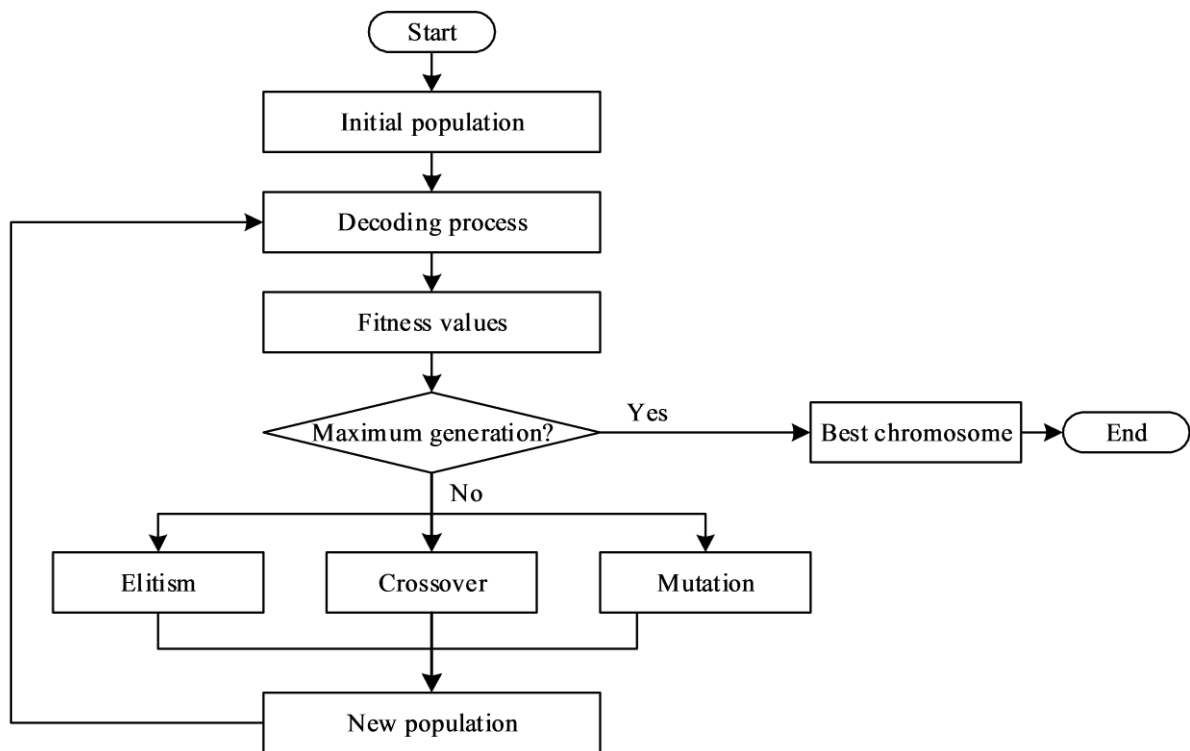
Figure 2. Flowchart of the proposed GA

Chromosome Representation

In this paper, a permutation-type chromosome adopted from Costa *et al.* [6] is applied to represent each chromosome. Each chromosome is divided into two segments. The first segment represents the DCs and the second one represents the warehouses. Thus, the length of the chromosome is the sum of possible DCs and warehouses. A locus in the chromosome segment represents a facility that may be established. In the permutation-type chromosome, the value in each locus represents the priority of the facility to be opened. A convention is used that number 1 is the highest priority and the next larger number will be less in priority. The values of each locus are unique across the two segments in a chromosome. Figure 3 illustrates a permutation-type chromosome with four possible DCs ad six possible warehouses to be opened. Table 1 shows the mapping value of locus to facilities priority.

Decoding Procedure

Facility priority mapping will be useful for the decoding procedure. The final goal of decoding process is to allocate demands to facilities. Once the demands are allocated, the total cost can be obtained through a known transport and facility building costs.

As the chromosome is divided into two segment, the decoding procedure can also be divided into two

steps. The first decoding step is done on the warehouse by allocating demands from regions to warehouses, and from warehouses to markets. The second decoding step is done on the DC segment where it allocate demands from warehouses to DCs, and from DCs to suppliers. The first decoding step can be described using flowcharts as shown in Figure 4 and Figure 5. The flowchart for demand allocation of the second step is similar.
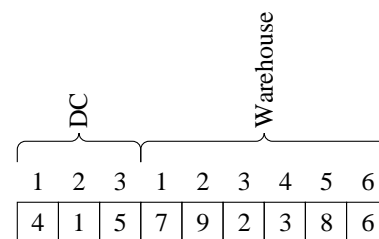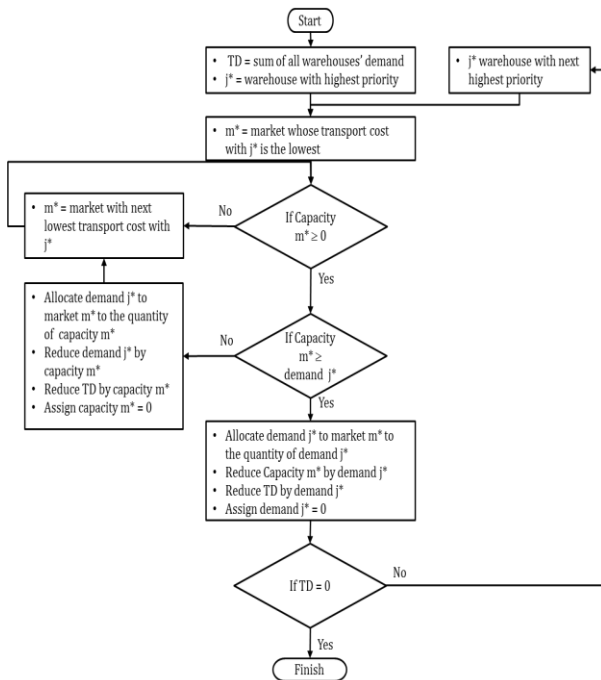
Figure 3. Illustration of the permutation-type chromosome

Table 1. Mapping value of locus to facilities priority

Facility	Priority	Facility to be opened
	1	DC 2
DC	2	DC 1
	3	DC 3
	1	Warehouse 3
	2	Warehouse 4
Warehouse	3	Warehouse 6
	4	Warehouse 1
	5	Warehouse 5
	6	Warehouse 2
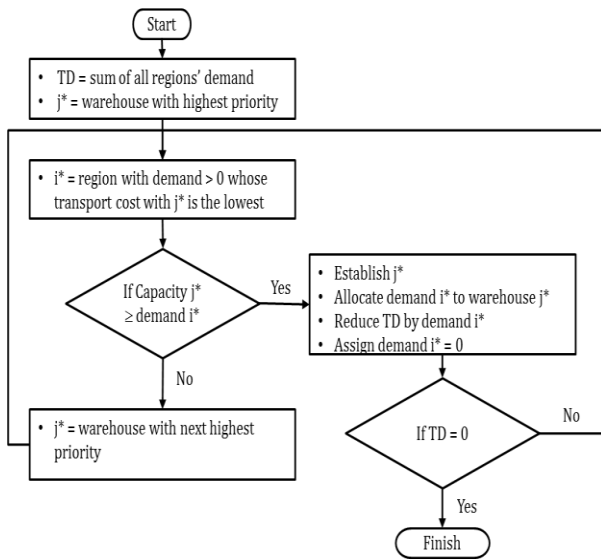
**Figure 4.** Flowchart of demand allocation from regions to warehouses.

**Figure 5.** Flowchart of demand allocation from warehouses to markets.

Genetic Algorithm Operators

Elitism. Elitism involves directly copying a number of the best solutions from the current population to the new population.

Crossover. Crossover is a method in genetic algorithm to mix best features of two candidate solution in order to find a better solution. This is done by mixing and exchanging the locus of two parent chromosome to obtain one or two new child chromosomes. The crossover method used in this paper is position-based crossover illustrated in Figure 6.
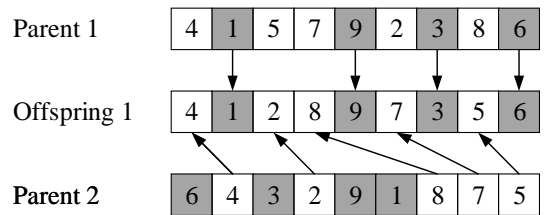
Figure 6. Illustration of position-based crossover
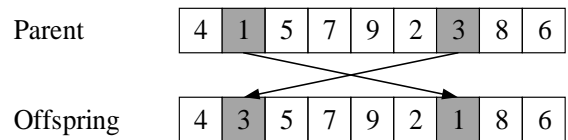
Figure 7. Illustration of swap operator

According to Costa *et al.* [6], the procedure for generating offsprings are described in the following steps.

1. Select a set of position from parent 1 at random.
2. Generate an offspring by copying from parent 1 the randomly selected genes into the corresponding position of the offspring.
3. Fill the blank genes in the offspring by adding not already included genes in the order as they appear in parent 2.
4. Repeat the steps 1-3 for a second offspring by starting from a set of positions randomly selected from parent 2.

Mutation. Mutation is done to a single chromosome to expand the solution space and escape a local optimum point. In the proposed GA, the swap operator is used. Based on the selected chromosome, the operator selects two genes randomly and exchanges their position each other. Figure 7 illustrates the swap operator.

Results and Discussions

The proposed GA is implemented in Visual Basic for Application (VBA) of Microsoft Excel 2013. To assess the performance of the proposed GA, computational experiments are performed using four hypothetical instances. The main characteristics of each instance are shown in Table 2.

The optimal solutions are obtained by solving the MILP model using LINGO 8. Both LINGO 8 and the proposed GA are executed on a computer with Intel Celeron 877 dual core 1.4 GHz processor, 4 GB RAM, and Windows 7 64-bit operating system. The GA parameters used are set as follows. The population size is 20. The percentages of the new individuals produced by elitism, crossover, and mutation are 10%, 60%, and 30%, respectively. The maximum number of generations is 100.

Table 2. Test instances

Instances	RG	WH	DC	IB	OB	SU_IB(1)	SU_IB(2)	SU_IB(3)	MA_OB(1)
1	3	3	3	3	1	1	2	2	2
2	5	5	5	3	1	1	5	5	5
3	10	10	10	3	1	1	3	3	3
4	30	30	30	3	1	1	3	3	3

RG: Number of regions, WH: Number of warehouses, DC: Number of distribution centers, IB: Number of inbound products, OB: Number of outbound products, SU_IB(1): Number of plants for inbound product 1, SU_IB(2): Number of plants for inbound product 2, SU_IB(3): Number of plants for inbound product 3, MA_OB(1): Number of markets for outbound product 1.

Table 3. Computational results

Instance	Optimal solution		GA solution	
	TC	CT	TC	CT
1	6,454	< 1 second	6,472	< 1 seconds
2	2,790,115	22 seconds	2,790,115	2 seconds
3	26,050	3.5 hours	26,155	4 seconds
4	Unknown	24 hours	88,697	25 seconds

TC: Total cost, CT = computation time

Table 4. Consistency for the proposed GA

Replication	Instance			
	1	2	3	4
1	6,472.0	2,790,115.0	26,179.0	88,847.0
2	6,472.0	2,790,115.0	26,155.0	97,038.0
3	6,472.0	2,790,115.0	26,179.0	93,619.0
4	6,472.0	2,790,115.0	26,173.0	91,611.0
5	6,472.0	2,790,115.0	26,155.0	88,697.0
6	6,472.0	2,790,115.0	26,155.0	92,554.0
7	6,472.0	2,790,115.0	28,185.0	94,674.0
8	6,472.0	2,799,121.0	26,155.0	94,092.0
9	6,472.0	2,790,115.0	26,155.0	97,708.0
10	6,472.0	2,790,115.0	26,179.0	94,744.0
Average	6,472.0	2,791,015.6	26,367.0	93,358.4
Standard deviation	0.0	2,847.9	638.9	3,027.6
Coefficient of variation	0.00	0.001	0.024	0.032

The proposed GA is run in 10 replications to obtain the best solution for each instance. Table 3 summarizes the computational results. From Table 3, it is shown that solving the MILP model can obtain the optimal solution only for three instances within 24 hours. For instance 4, solving the MILP model cannot find a feasible solution until 24 hours. But, using the proposed GA, the solution can be found within 25 seconds. Based on the optimal solutions obtained (instance 1-3), the gap in the total cost is very small (0.23%, on average).

Figure 8 shows the convergence of the proposed GA based on instance 1 where it shows the best total cost for each generation. It is known that GA is a probabilistic metaheuristic approach. Therefore, each replication may produce a different solution. In terms of total cost, based on coefficients of variation, the consistency of the proposed GA is shown in Table 4.

It is shown that the coefficients of variations are close to 0 (based on 10 replications) and it demonstrates the consistency of the proposed GA. It is also observed that the bigger problem size will have a higher coefficient of variation.
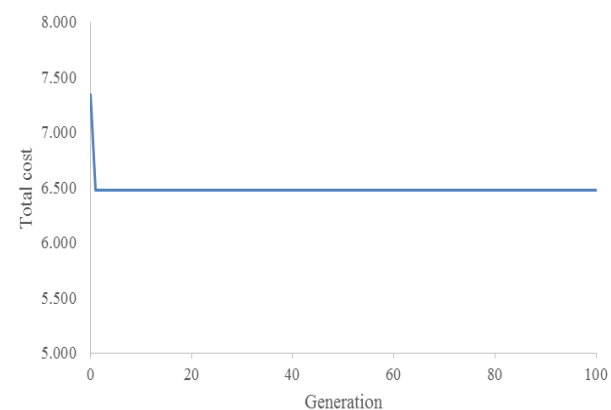
Figure 8. Convergence of the prposed GA

Conclusions

This paper has presented the facility location problem in a supply chain network consisting of inbound and outbound product flows. The solution approach based on genetic algorithm (GA) has been proposed. A computational experiment has been conducted to evaluate the performance of the proposed GA. The solutions of the proposed GA are compared to the optimal solutions obtained by solving the MILP model.

The comparison shows that there is a small gap (0.23%, on average) between the proposed GA and MILP model in terms of the total cost. In addition, the proposed GA can obtain the solution for the instance that cannot be obtained by solving the MILP model. Therefore, the proposed GA can be applied for solving large problems where they exist in the real situation. In addition, the proposed GA provides consistent results. In term of the total cost, it is demonstrated that coefficients of variation are closed to 0.

References

1. Suprayogi, Bahagia, S. N., and Thaddeus, Y., Location Problem in a Supply Chain Network with Inbound and Outbound Product Flows, *Proceedings of 6th International Conference on Operations and Supply Chain Management*, 2014, pp. 1353–1363, Bali.

2. Min, H., and Zhou, G., Supply Chain Modeling: Past, Present and Future, *Computers and Industrial Engineering* 43, 2002, pp. 231–249.

3. Melo, M. T., Nickel, S., and Saldanha-da-Gama, F., Facility Location and Supply Chain Management–A Review, *European Journal of Operational Research* 196, 2009, pp. 401–412.

4. De Rosa, V., Gebhard, M., Hartmann, E., and Wollenweber, J., Robust Sustainable Bi-Directional Logistics Network Design under Uncertainty, *International Journal of Production Economics* 145, 2013, pp. 184–198.

5. Zhang, Z. -H., Li, B. -F, Qian, X., and Cai, L. -N., An Integrated Supply Chain Network Design Problem for Bidirectional Flows, *Expert Systems with Applications*, 41, 2014, pp. 4298–4308.

6. Costa, A., Celano, G., Fichera, S., and Trovato, E., A New Efficient Encoding/Decoding Procedure for the Design of a Supply Chain Network with Genetic Algorithms, *Computers and Industrial Engineering*, 59, 2010, pp. 986–999.