

Bi-objective freight scheduling optimization in an integrated forward/reverse logistic network using non-dominated sorting genetic algorithm-II

Taufik Djatna^{a*} and Guritno A. M. Amien^b

^aPost Graduate Program, Department of Agro-industrial Technology, IPB University, Bogor, Indonesia

^bDepartment of Agro-industrial Technology, IPB University, Bogor, Indonesia

CHRONICLE

Article history:

Received June 15, 2019

Received in revised format:

June 20, 2019

Accepted July 27, 2019

Available online

July 27, 2019

Keywords:

Bi-objective optimization

Freight scheduling

Integrated forward/reverse

logistic network

Non-dominated sorting genetic

algorithm-II

ABSTRACT

Simultaneous products distribution and items retrieval in an integrated forward/reverse logistics network faces a complex freight-scheduling problem due to the constraints involved. In the high to intermediate network level, the problem usually exists in the form of single stop transportation. To reach a higher level of performances, there is a need to model and optimize the freight schedule. This research proposes a model to optimize a freight-scheduling problem. The proposed model of this paper based on Non-dominated sorting genetic algorithm-II is formulated to solve a conflicting bi-objective optimization and optimizes a real-world case study. A solution from the model demonstrates the solution interpretation in the form of delivery schedule, distribution as well as retrieval route, and vehicle assignment. Moreover, the solutions are also comparable to some current manual solution by its similarity. The results show that the model was capable of generating feasible solutions while satisfying all of its constraints.

© 2020 by the authors; licensee Growing Science, Canada.

1. Introduction

Freight scheduling is a series of transportations of a bulk/large quantity of goods in a limited time. Freight scheduling problem is considered as a sub-discussion of freight management that involves vehicle routing, vehicle scheduling and dispatching, freight network flow, freight consolidation, etc. (Gudehus & Kotzab 2009). Freight scheduling is important because it manages transportation of items in a logistic network. Transportation itself occupies one third of the amount of the logistics cost and hugely influences the performance of logistics system. Therefore, optimization of a freight schedule is important to reduce the overall logistics cost and enhances the logistic system's performance (Parkhi et al., 2014; Tseng et al., 2005). Many real world problems are recently involved with optimization of multiple conflicting objectives (de Oliveira & Saramago, 2010). Hence freight schedule optimization is also preferably to be optimized with more than one objective. For example, minimizing transportation cost and maximizing order responsiveness. In this case, there are two conflicting objectives, thus the optimization is called bi-objective optimization.

Freight schedule optimization exists in forward and reverse logistics. Forward logistics is described as the processes (including planning, implementing, and controlling) involved in the movement of materials (including raw materials, in-process inventory, finished goods, and related information) from the point of origin towards the point of consumption. The opposite term of it is reverse logistics, which

* Corresponding author.

E-mail address: taufikdjatna@apps.ipb.ac.id (T. Djatna)

is described as processes involved in the movement of materials from the point of consumption to the point of origin for the purpose of recapturing or creating value or proper disposal (Rogers & Tibben-Lembke, 1999). Enterprises are interested in implementing reverse logistics because it is one of the most common driving force, that is economic factor. A reverse logistics program might bring direct benefit to companies by decreasing the use of raw materials, adding value with recovery, or reducing disposal costs (de Brito & Dekker, 2003). Freight scheduling optimization in any logistics network is a critical problem to solve, be it forward or reverse logistics. However, optimization in a separate forward and reverse logistics network may result in a sub-optimal solution. Therefore, an Integrated Reverse Logistic Problem (IRLP) was introduced. IRLP is a logistics network type where the forward and reverse logistic are designed or managed in an integrated manner in terms of facility, transportation route, or transportation schedule. The integration was performed to evade the sub-optimality (Pishvaei et al., 2009).

In order to optimize a freight schedule in an IRLP, a Graph Theory is potentially used to model the network. The node or vertex is used to represent the facility while the arc or edge is used to represent the shortest route connecting two facilities. The optimization in freight schedule is performed by determining the lowest cost route (similar to Minimum Spanning Tree/MST) and also the quantity of distribution and retrieval. Classical minimum spanning tree techniques such as Kruskal's algorithm, Boruvka's algorithm, and Prim's algorithm are not suitable for a multi objectives optimization that is addressed in this research. Furthermore, these techniques failed to solve a large scale problem that usually involve multi-dimensionality which is addressed in this research in the form of multi products and multi-capacitated vehicles. However, an advanced optimization approach, such as Non-dominated Sorting Genetic Algorithm-II (NSGA-II) is suitable for multi objectives optimization and capable of solving a large scale problem (Rao & Savsani 2012). NSGA-II was proposed to solve the high computational complexity, lack of elitism, and specifying of the sharing parameter of NSGA. In NSGAII, a selection operator is designed by creating a mating pool to combine the parent population and offspring population. Non-dominated sort and crowding distance ranking are also implemented in the algorithm. Therefore, it is potentially used to solve the problem on this research.

This research tried to model, to optimize a vehicle routing, and to network flow in an IRLP. The research focus is only in the high to intermediate network level, which is between manufacturer and distribution centers. The objectives of this research is to formulate a transportation optimization model, which utilizes a Non-dominated Sorting Genetic Algorithm II (NSGA-II), and to optimize a given case study using the proposed model.

The remainder of this paper is organized as follows. In section 2, the literature reviews related to the field of research are discussed. In section 3, the research problem and also its assumptions are described. Then in section 4, a mathematical model and optimization approach are proposed. In section 5, a Java code that is implemented to solve the problem is elaborated and in section 6, the result and discussion of applying the code to a case study are presented. The model is further interpreted for solutions similarity, advantages, disadvantages, and managerial impacts are presented in section 7. Finally, the conclusion of this research and recommendation for future works are presented in section 8.

2. Literature review

Transportation system is a collection of components or elements that work together to provide a safe and efficient movement of people and goods. A transportation network often is represented using Graph Theory. Graph is a pair $G = (V, E)$ where V is the set of vertices (or nodes, or points) of the graph G , and E is the set of edges (or arcs, or lines) formed by pairs of vertices (Diestel 2005). Various studies have been conducted on the field of graph theory for the last few years. Likaj et al. (2013) presented the use of Dijkstra's and Kruskal's algorithm to find the shortest path and minimum spanning tree which minimized the shipment cost. Barwaldt et al. (2014) studied the use of graph theory for the

implementation of bike lane in a small town. They found out that by using the graph theory, the bike lane was successfully generated by minimizing the cost and time of implementation. Price and Ostfeld (2014) also presented the use of Successive Shortest Path (SSP) algorithm to solve the minimum-cost flow problem for a water system. They compared the results generated from the SSP algorithm with the results generated from linear programming and reported that by using the SSP algorithm, the water would be held for fewer hours in the water tanks before consumption, which yields to improve the water quality dispatch to consumers. The use of graph theory on reverse logistics was presented by Agrawal et al. (2016). They attempted to find the various disposition alternatives and developed an approach for the selection of best disposition alternative using graph theory and matrix approach. They proved that the proposed approach was capable of selecting the best disposition alternative in a case study. Recently, Démare et al. (2017) presented the use of a dynamic graph to model and simulate logistics system. They claimed that the proposed model might be implemented to simulate many logistics systems. The graph theory that have been explained above only worked well in not complex scema (Guidice 2013). The optimization of transportation system might be performed using classical or advanced techniques. The utilization of genetic algorithm as one of advanced techniques in the field of transportation system have been researched quite a lot. Siregar (2012) developed a model to optimize a vehicle routing problem without time windows in a forward logistics network using basic genetic algorithm. Zaki et al. (2012) developed an efficient approach to solve a transportation, assignment, or transshipment problem in a forward logistics context using hybrid genetic algorithm with local search algorithm. Cataruzza et al. (2013) proposed a procedure that outperforms some common algorithm to solve a Multi Trip Vehicle Routing Problem (MTVRP) in a forward logistic network. The proposed procedure consisted of splitting procedure, genetic algorithm, and local search.

Numerous studies have been conducted in the field of freight scheduling that was related to transportation system in both forward or reverse logistics and even in the integrated logistics network issues. Fleischmann et al. (2001) developed a model to integrate reverse logistics network design in case of facility location's determination to an existing multi echelons logistic structures. Lee and Dong (2008) developed a method to efficiently solve the location-allocation and network flow in a multi echelon IRLP with single product multiple components using Tabu search approach. Khajavi et al. (2011) proposed a model to optimize a capacity and location problem in a multi echelon IRLP with single product using branch and bound algorithm. Baumik (2015) designed a formulation of minimum cost in routing reverse logistics form warehouse to retail stores. He applied ILP (integer linear programming) while others applied MILP (mix integer linear programming) or MIP (mix integer programming) (Fazlollahtabar, 2018). But this method only worked for not very large problems. Lastly, Dondo and Mendez (2016) presented a framework to optimize network flow operational planning in a multi echelon IRLP with single product using a column-generation based decomposition approach. From the previous studies mentioned, it is known that optimization of vehicle routing and network flow for the IRLP was rarely performed using evolutionary algorithm such as genetic algorithm. Not to mention that most cases only considered single product. Moreover, the use of graph theory was mainly implemented by using a classical algebraic optimization approach which is more suitable for limited variables and known functions. Therefore, this research was performed to accommodate a multi products cases while utilizing an NSGA-II algorithm as the optimization approach. This optimization approach was deployed because it is popular, fast, reliable, and capable to address a multi objective optimization. Since the problem addressed in this research has two objectives to be optimized, the utilization of NSGA-II approach is a sensible choice.

3. The proposed methodology

3.1. The problem statement and assumption

The problem addressed in this research is the optimization of the route as well as quantity of simultaneous distribution (forward logistic activity) and retrieval (reverse logistic activity). It also

required the determination of vehicle assignment in a single time windows. Moreover, contribution of this work was focused in the high to intermediate transportation network level, which is between factories and distribution centers. The problem discussed has characteristics of single echelon freight transportation, single stop, single manufacturing site, multi products, and multi capacitated vehicles. Furthermore, the problem is consisted of two conflicting objectives of transportation cost and order responsiveness, both in forward logistics, as well as in reverse logistics. These two objectives were determined from the transportation system requirements as a case study, which elaborated, in the next section.

As illustrated in Fig. 1, in a forward logistics network, the products (e.g. beverages in a Returnable Glass Bottle, abbreviated as RGB) are distributed to satisfy demand at day x from a set of DCs. Notation x refers to the day where the distribution and retrieval are optimized. The order responsiveness for forward logistics refer to the total number of products distributed per total demand at day x . In the reverse logistics network, the retrieved items (e.g. empty Returnable Glass Bottle) are transported from a set of DCs to manufacturer in order to satisfy the forecast of production requirements at day y . Notation y refers to the day where the retrieved items are needed for production. The order responsiveness for reverse logistic refers to the total amount of item retrieved per total production requirement at day y . Order responsiveness is useful to understand how well a freight schedule reacts to the change in products demand or retrieved items requirement.

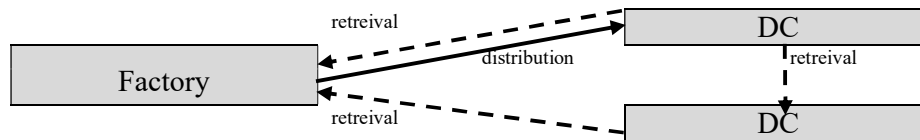


Fig. 1 Illustration of distribution and retrieval in IRLP.

The problem in this research only allowed a DC to be visited exactly once for the same vehicle. However, the DC also allowed to visit by multiple vehicles a day. This means that if vehicle-1 visits DC-A, then it can visit the other DCs except DC-A on the same day. On the other hand, DC-A might still be visited by the other vehicles. The time constraint for this problem is in the form of single time windows and only applicable for the forward logistics. The reason is because from the preliminary study of the real-world case (used as the case study later), it was known that the time constraint for the reverse logistics was very lax. Thus, it was nearly impossible for the delivery in the reverse logistics to be tardy.

3.2 Mathematical model and optimization approach

3.2.1 Proposed notations and mathematical model

A mathematical equations as the problem representation and the solutions similarity were formulated. The model was consisted of two objectives and nine constraints. The optimization approach (NSGA-II) would produce multiple solutions where each solution has a set of variables. Hence, it is best to include the solution and variable's indices in the model. For each solution, the list of indices, decision variables, and parameters of the proposed model is presented as in Table 1. Furthermore, the goals and constraints of the proposed model is presented as in the equations below this table. The formulation of solution's similarity was also presented to be used in the section 5 on this paper.

Table 1

List of notations

| No. | Notation | Description | Range |
|----------------|----------|------------------------------|-------------------|
| <i>Indices</i> | | | |
| 1. | v | Vehicle's indices | $\{1, \dots, v\}$ |
| 2. | p | Product's indices | $\{1, \dots, p\}$ |
| 3. | r | Retrieved item's indices | $\{1, \dots, r\}$ |
| 4. | i | Factory as a node's indices | $\{1, \dots, i\}$ |
| 5. | j | DC as a node's indices | $\{1, \dots, j\}$ |
| 6. | x | Day's indices | $\{1, \dots, x\}$ |
| 7. | c | Vehicle capacity's indices | $\{1, \dots, c\}$ |
| 8. | s | Loading server's indices | $\{1, \dots, s\}$ |
| 9. | y | order of departure's indices | $\{1, \dots, y\}$ |
| 10. | o | Solution's indices | $\{1, \dots, o\}$ |
| 11. | a | Variable's indices | $\{1, \dots, a\}$ |

Where all index are integer

Decision variables

| | | |
|----|-----------------|--|
| 1. | d_{vpj}^{xo} | Amount of product p distributed from node i to node j by vehicle v at day x |
| 2. | e_{vrji}^{xo} | Amount of retrieved item p transported from node j to node i by vehicle v at day x |

Parameters

| | | |
|-----|----------------|---|
| 1. | SP_{pi}^x | Stock of product p on node i at day x |
| 2. | SR_{rj}^x | Stock of retrieved item p on node j at day x |
| 3. | DID_{pji}^x | Demand for product p at day x from node j to node i |
| 4. | Cap_c | Constant value of vehicle's capacity c |
| 5. | k_c | Constant value of fleet size for vehicle's capacity c |
| 6. | Dep_c | Number of vehicles deployed for capacity c |
| 7. | CD_{ij} | Cost of distribution from node i to node j |
| 8. | CR_{ji} | Cost of retrieval from node j to node i |
| 9. | CB_{ij} | Base cost of transportation between node i and node j |
| 10. | CBD_{jj} | Base cost of transportation between two node j |
| 11. | θ | Weighting factor for the order responsiveness of forward logistic |
| 12. | $(1 - \theta)$ | Weighting factor for the order responsiveness of reverse logistic |
| 13. | dT_s^y | Departure time of order y at server s |
| 14. | lT | Loading time |
| 15. | tT_{ij} | Transportation time from node i to node j |
| 16. | Z_1 | Transportation cost |
| 17. | Z_2 | Order responsiveness |
| 18. | n | Number of variable(s) in a solution to be calculated in solution's similarity |
| 19. | Sim_n^o | Similarity between solution o and solution reference for n variable(s) |

Mathematical model:

$$\min Z_1 = \sum_v \sum_j \left(\sum_p (CD_{ij} * d_{vpj}^{xo}) \right) + (CB_{ij}) + \sum_v \sum_j \left(\sum_r (CR_{ji} * e_{vrji}^{xo}) \right) + (CB_{jj} + CBD_{ij}) \tag{1}$$

$$\max Z_2 = \left[\theta * \frac{\sum_v \sum_j \sum_p d_{vpj}^{xo}}{\sum_j \sum_p DID_{pji}^x} \right] + \left[(1 - \theta) * \frac{\sum_v \sum_j \sum_r e_{vrji}^{xo}}{\sum_j \sum_p DID_{pji}^{x+const}} \right] \tag{2}$$

subject to

$$SP_{pi}^x \geq \sum_v \sum_j d_{vpij}^{xo} \quad (3)$$

$$SR_{rj}^x \geq \sum_v e_{vrji}^{xo} \quad (4)$$

$$Z_2 \leq \text{constant number} \quad (5)$$

$$\sum_v d_{vpij}^{xo} \geq DID_{pji}^x \quad (6)$$

$$\sum_v \sum_j e_{vrji}^{xo} \geq DID_{pji}^{x+const} \quad (7)$$

$$\sum_p d_{vpij}^{xo} \leq Cap_c \quad (8)$$

$$\sum_r e_{vrji}^{xo} \leq Cap_c \quad (9)$$

$$Dep_c \leq k_c \quad (10)$$

$$dT_{ij}^{y-1} + (lT * d_{vpij}^{xo}) + tT_{ij} \leq \text{Max Estimated Time of Arrival (ETA)} \quad (11)$$

The first objective (1) calculates the total transportation cost, including cost of distribution/item, cost of retrieval/item, and basic cost for each delivery. The second objective (2) calculates the total responsiveness both from forward logistics and reverse logistics with a weighting factor. Denominator at responsiveness of reverse logistics is the total daily demand at day ($x + \text{constant number}$). Let $\text{constant number} = n_1 + n_2$, where n_1 is gap in day(s) between the day of the items is retrieved and the day of its intended used for production, and n_2 is gap in day(s) between the day ($x + n_1$) and the day the products produced from day ($x + n_1$) is intended to distributed. Therefore, the total daily demand at day ($x + \text{constant number}$) is equal to the retrieved items requirement for production at day ($x + c_1$), which in turn equals to the forecast of total demand of products at day ($x + c_1 + c_2$). Eq. (3) and Eq. (4) ensure that the total products distributed and the total items retrieved will not exceed the available stock. Eq. (5) is optional but much recommended to limit the second objective value. The total distribution for each product and each DC is ensured to satisfy the demand based on Eq. (6). Similarly, Eq. (7) ensures that the total item retrieved will satisfy the retrieved item requirement. Eq. (8) and Eq. (9) limit the total distribution or retrieval for each vehicle to be less than equal to vehicle capacity. Eq. (10) ensures the total vehicle deployed must not exceed the fleet size for each capacity. Lastly, Eq. (11) makes sure that the distribution will not exceed the time windows.

3.2.2 Proposed optimization approach

The problem defined above required an exhausting search step for the optimization since the amount of possible solutions is very massive and the function is not well known. During data acquisition of observation, there are 20 vehicles available to execute distributions and retrievals between a manufacturer and 10 DCs. Moreover, there are 25 possible amounts of distribution or retrieval including not transporting anything (zero) for 4 products and 3 retrieved items. The amount of possible solutions will be equal to a series of permutation with repetition of ${}_{n_1}^{r_1}P_1 \times {}_{n_2}^{r_2}P_2 = {}_{25}^{(4+3) \times 20}P_1 \times {}_{10}^{2 \times 20}P_2 = 7.6 \times 10^{238}$. Notation n_1 represents possible elements of distribution/retrieval, n_2 represents possible elements of DCs, r_1 representw ordered arrangements length from 4 products, 3 retrieved items, and 20 vehicles, and r_2 represent ordered arrangements length from 20 vehicles for distribution and 20 vehicles for retrieval. Optimization with meta-heuristic approach such as genetic algorithm is suitable for optimization with large search space. Moreover, since the proposed model is optimizing two conflicting objectives, there is a need to use genetic algorithm that is capable of addressing multi

objectives optimization. Therefore, a Non-Dominated Sorting Genetic Algorithm II (NSGA-II) is deployed on this research because it is a fast, reliable, and easy to implement algorithm. NSGA-II is developed by Deb et al. (2002), which utilizes a fast non-dominated sorting approach to determine the Pareto front, a crowded comparison approach to maintain the diversity of solutions, and an elitist principle.

Chromosomes are representation of the solutions for the problem addressed in the optimization using NSGA-II. The chromosomes structure proposed in this research have been encoded by integer value and consisted of two group of sub-chromosomes. Fig. 2 illustrates the representation of chromosomes. The first group was composed of sub-chromosomes of forward logistics, such as sub-chromosomes of distribution's amount for each product $P_p = \{P_1, P_2, \dots, P_n\}$; $n \geq 1$; $n \in p$ and sub-chromosomes of node destination (Distribution Center as destination node). The second group was composed of sub-chromosomes of reverse logistics, such as sub-chromosomes of retrieval's amount for each retrieved item $R_r = \{R_1, R_2, \dots, R_n\}$; $n \geq 1$; $n \in r$ and sub-chromosomes of node origin (Distribution Center as origin node). Each sub-chromosome has a set of genes (variables) equal to the total fleet size ($\sum_c k_c$ in the mathematical model). Sub-chromosomes of distribution or retrieval's amount has n allele (equal to the number of possible amount of distribution/retrieval). While sub-chromosomes of node destination or node origin has m allele (equal to the number of existing nodes).

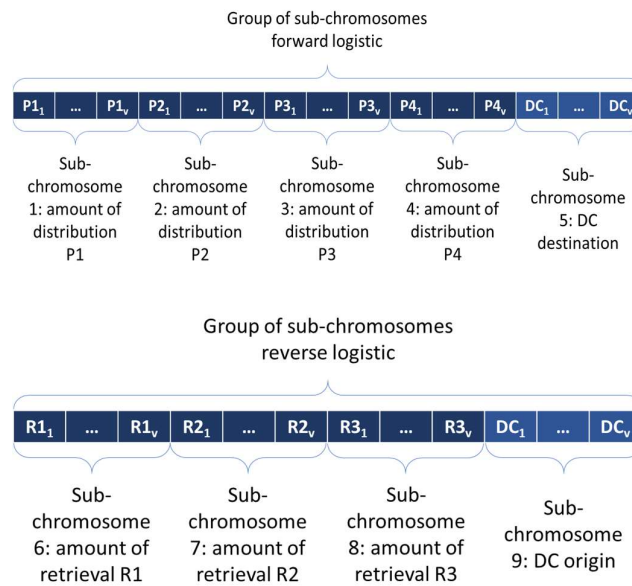


Fig. 2. Example of chromosome representation with 4 products and 3 retrieved items

4. Code impementation

The proposed model was structured and then gradually coded in Java language using Eclipse IDE with the help of Multi-Objectives Evolutionary Algorithm (MOEA) framework version 2.12 (Hadka 2017). MOEA framework is an open source Java library used for developing multi-objectives optimization using evolutionary algorithm and other general purpose optimization algorithm. The code was structured in a set of classes. There were ten classes outside the MOEA framework. These classes act as the template for Java objects that represent the actual objects in the transportation system and objects in the optimization approach. Fig. 3 illustrates how the code was structured in a set of classes. The notations and equation of mathematical model (as explained in Table 1 and the equations below it) were constructed as in the class diagram above. The indices, decision variables, and parameters were constructed as attribute for the classes. Meanwhile, the equations were constructed as methods in a class called as constraintFitness. The methods from the other classes outside the MOEA framework generally utilize setter and getter method to assign value of attributes in the object. The constructor

such as `initiateFuel ()` and `Solution ()` were used to create an object from the respective class. The class `Distribution Center`, `Vehicle`, `Product`, `Retrieved Item`, `Fuel`, and `Date` were constructed as a template based on the actual objects in a transportation system. The objects that were generated based on the class `Distribution Center`, `Vehicle`, and `Fuel` were used to construct a composite object on the class `Network`. The object's parameters in class `Demand` were based on the composite object from class `Date` and `Product`. Similarly, the object's parameters in class `Stock` were based on the composite object from class `Date`, `Product`, and `Retrieved Item`.

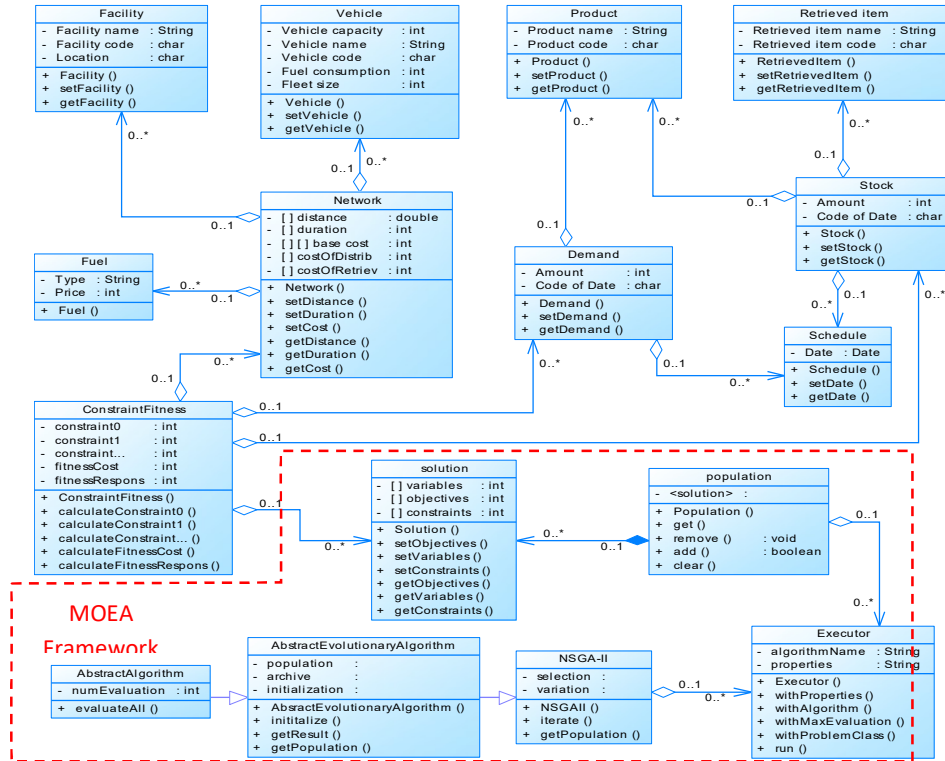


Fig. 3. Proposed class diagram for Java coding

The class `ConstraintFitness` was used to calculate the constraints' value and objectives' value of each object from class `Solution`. This class used the objects from class `Solution`, `Network`, `Demand`, and `Stock` to calculate the value of parameters constraints and objectives then set these parameters' value to the respective object `Solution`. The objects based on class `solution` were used by MOEA framework through the NSGA-II algorithm. A single object from class `Population` was composed by many objects from class `solution`. These objects were going through selection, mutation, and crossover based on the NSGA-II Algorithm. The `Executor` is a class from MOEA framework that configures and executes the algorithm. Since the model proposed in this research utilized NSGA-II, this class would refer to `NSGAI` class as the value of variable `algorithmName`. This class would call `NSGAI` class and used the method to execute the optimization. `NSGAI` class extends the `AbstractEvolutionaryAlgorithm` in order to use some general evolutionary operator such as initializing population, crossover, selection, etc. The `AbstractEvolutionaryAlgorithm` class in turn extends `AbstractAlgorithm` to use the method for evaluating the solution (chromosome) in `population` class. The input data for optimization were provided in two ways. The first one was provided from a database, while the second one was provided directly from the code body. This would make the input process for the case study becomes easier. The database was written in MySQL format (Oracle 2017). The NSGA-II in MOEA Framework generally utilize binary tournament selection and deploy Simulated Binary Crossover (SBX) and Polynomial Mutation (PM) only in case the chromosomes are encoded by integer or real value. In SBX, the offspring distribution of single point crossover with binary encoding is simulated on a real valued decision variables (Deb & Agrawal 1994). Similar to SBX, PM simulate the offspring distribution of

bit-flip mutation with binary encoding on a real valued decision variables (Deb & Goyal 1996). Both SBX and PM has attribute of rate and distribution index. Rate means the probability of the operator (SBX or PM) is applied to a decision variable. Distribution index means the offspring distribution's shape. Larger value means closer offspring to the parent (default value for SBX is 15 and for PM is 20) (Hadka 2011). Since MOEA framework only allowed minimization, the second objective of the model, which is to maximize the order responsiveness, was coded in a negative value to change the maximization problem into minimization problem.

5. Implementation of the proposed model in a case study

In the following section, as a case study to evaluate the solution, an observation on the operation of PTY was taken for consideration. PTY is one of leading companies from Indonesia that is dealing with ready to beverages, mainly tea and its derivative. These beverages are contained in PET bottle, carton packaging, or Returnable Glass Bottle (RGB). For the case study, there are four products in the form of ready to drink tea in RGB container and three retrieved items in the form of empty RGB container.

5.1 Test problem

The freight network of the problem was represented using a Graph Theory as mentioned before. The graph (G) has 13 vertices (V) representing the manufacturer and Distribution Centers (DCs) of PTY. The total fleet size of the sample company were 26 with detail of each capacity provided in Table 2. The vehicle number was determined according to the fleet size for each capacity. That means vehicles number 1 to 18 represent the vehicle with capacity of 1,200; vehicle number 19 represent vehicle with capacity of 840; vehicle number 20 represent vehicle with capacity of 720; and vehicle number 21 until 26 represent vehicle with capacity of 400.

Table 2

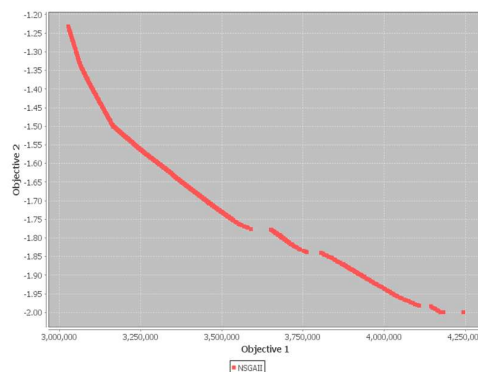
Fleet size for each vehicle capacity

| Capacity (crate) | 1200 | 840 | 720 | 400 |
|-------------------|------|-----|-----|-----|
| Fleet Size (unit) | 18 | 1 | 1 | 6 |

The other data for optimization were provided in the form of database and in the code body. The problem was solved using the proposed model with 600 individuals as initial populations, through 1,600,000 generations, with PM rate 10/total variable, SBX rate of 1.0, and PM as well as SBX distribution index of 5.

6.2 Model's output

According to two objective function, described in formula (1) and formula(2), it was observed that the model generated 402 feasible and optimum solutions with representation on the objective spaces as in as in Fig. 4.



Min transportation cost (rupiah)

Fig. 4 Pareto front at 300,000th generation

From this figure, the solutions were forming a Pareto frontier. This was aligned with the theory of evolutionary algorithm, which stated that the feasible solutions in an evolutionary algorithm would form a front if viewed together on the objective spaces (Kacprzyk & Pedrycz 2015). The rightmost solution had the first objective value of Rp 4,244,558 and second objective received a value of 1.999. While the leftmost solution had the first objective value of Rp 3,026,897 and the second objective received a value of 1.231.

Table 3

Detail of solution number 3

| Sub-chromosome | Gene's value | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|--------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|----|---|----|----|---|---|
| mosome | | | | | | | | | | | | | | | | | | | | | | | | | | |
| me | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 2 | 14 | 16 | 22 | 13 | 3 | 14 | 1 | 5 | 6 | 4 | 10 | 14 | 19 | 21 | 8 | 3 | 10 | 4 | 6 | 1 | 2 | 4 | 3 | 1 | 0 |
| 2 | 20 | 5 | 2 | 1 | 4 | 16 | 1 | 4 | 4 | 16 | 2 | 0 | 7 | 2 | 3 | 13 | 19 | 2 | 0 | 0 | 4 | 0 | 0 | 5 | 1 | 0 |
| 3 | 1 | 1 | 4 | 0 | 2 | 4 | 9 | 0 | 1 | 1 | 1 | 2 | 2 | 0 | 0 | 3 | 1 | 4 | 3 | 3 | 2 | 0 | 0 | 0 | 0 | 5 |
| 4 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 4 | 0 | 1 | 2 | 2 | 1 | 3 | 0 | 0 | 1 | 5 | 3 | 0 | 1 | 3 | 0 | 0 | 0 | 0 |
| 5 | 11 | 1 | 0 | 1 | 0 | 11 | 9 | 0 | 0 | 11 | 0 | 3 | 11 | 1 | 9 | 11 | 11 | 1 | 3 | 0 | 11 | 0 | 0 | 11 | 0 | 0 |
| 6 | 20 | 18 | 21 | 9 | 0 | 18 | 16 | 16 | 10 | 11 | 21 | 22 | 8 | 20 | 12 | 14 | 15 | 18 | 10 | 5 | 6 | 7 | 3 | 4 | 2 | 1 |
| 7 | 0 | 5 | 2 | 6 | 0 | 4 | 0 | 0 | 1 | 9 | 1 | 0 | 2 | 3 | 0 | 5 | 5 | 3 | 3 | 7 | 0 | 0 | 0 | 2 | 1 | 1 |
| 8 | 1 | 1 | 0 | 2 | 24 | 0 | 0 | 8 | 13 | 4 | 1 | 2 | 7 | 1 | 11 | 0 | 0 | 3 | 3 | 2 | 1 | 0 | 5 | 1 | 0 | 4 |
| 9 | 1 | 9 | 9 | 1 | 11 | 9 | 9 | 11 | 11 | 3 | 9 | 12 | 11 | 12 | 11 | 0 | 0 | 12 | 4 | 9 | 0 | 9 | 11 | 11 | 0 | 0 |

Sub-chromosome 1 to 5 represent the product and sub-chromosome 6 to 8 represent the retrieved item. Gene's value in these sub-chromosomes depict amount of each freight. The rest of the sub-chromosomes assumed as distribution center (DC). Gene's value in these sub-chromosomes depict the DC number. Each gene represents the vehicle number accordingly. For instance, based on the first column shown in bold numbers in Table 3, vehicle 1 must distribute 2×50 product 1, 20×50 product 2, 1×50 product 3 and 1×50 product 4 with destination node, according sub-chromosome 5, is DC number 11. We named the product 1 as RGB1, product 2 as RGB2, product 3 as RGB3. Then vehicle 1 must go to DC number 1 (according to sub-chromosome 9) to retrieve 20×50 RGB 1, zero RGB 2 and 1×50 RGB 3 before going back to manufacturer. The total duration for loading time and distribution time then are calculated with three servers for freight loading process. Table 4 provides a complete freight schedule representation with each vehicle's Estimated Time of Arrival (ETA).

Table 4

Distribution schedule representation

| Vehicle | Amount of distribution (crates) | DC distribution | Sequence | Estimated Time of Arrival (hh:mm) |
|---------|---------------------------------|-----------------|----------|-----------------------------------|
| 14 | 1200 | 1 | 1 | 11:09 |
| 4 | 1150 | 1 | 1 | 11:07 |
| 2 | 1050 | 1 | 1 | 11:02 |
| 18 | 1050 | 1 | 2 | 12:02 |
| 12 | 700 | 3 | 2 | 11:51 |
| 19 | 500 | 3 | 2 | 11:36 |
| 1 | 1200 | 11 | 3 | 11:48 |
| 10 | 1200 | 11 | 3 | 11:28 |
| 13 | 1200 | 11 | 3 | 11:13 |
| 15 | 1200 | 11 | 4 | 12:48 |
| 16 | 1200 | 11 | 4 | 12:28 |
| 17 | 1200 | 11 | 4 | 12:13 |
| 6 | 1200 | 9 | 5 | 13:25 |
| 7 | 1200 | 9 | 5 | 13:05 |
| 21 | 400 | 11 | 5 | 12:33 |
| 24 | 400 | 11 | 6 | 14:08 |

In addition, graphical description for the distribution and retrieval explained above is easily depicted in Fig. 5 and Fig. 6. The route for forward logistic (distribution) named graph G_1 was presented as in Fig. 5. The graph G_1 has 13 vertices (represent by number in a circle) $V = \{0, 1, \dots, 12\}$. The vertice

zero as the manufacturer and the other vertices as the DCs. This Graph has only four edges $E = \{A, B, C, D\}$. The edges represent the route for which vehicle number goes through and illustrated as the black arrow. Black arrow means the distribution route. According to this figure, there were 24 vehicles used for distribution. Only vehicle number 25 and 26 were not used. Vehicle number 1 through 24 went through five different routes with 12 vehicles through route A, four vehicles through route B, six vehicles through route C, and two vehicles through route D. For example in a blue rectangle area at Fig. 5, vehicle number 2, 4, 14, and 18 were transporting products to DC 1. The vertices that were not connected on the graph depicted in Fig. 5 means that there were no products transported to these vertices. This was happening because these vertices do not have products demand so the model does not distribute any products to it.

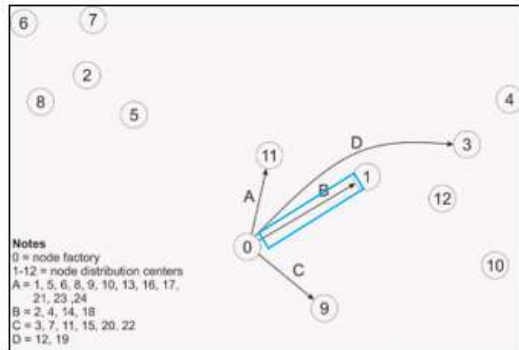


Fig. 5 Route representation for forward logistic

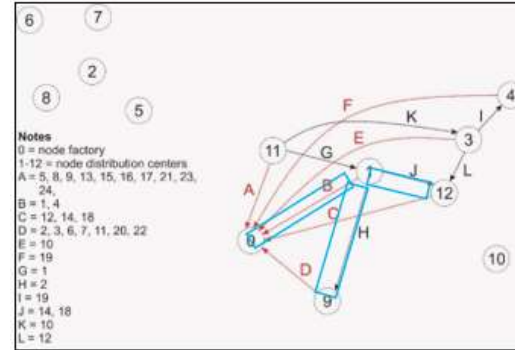


Fig. 6 Route representation for reverse logistic

The route for reverse logistic (retrieval) named graph G_2 was presented as in Fig. 6. This Graph has 12 edges $E = \{A, B, C, \dots, L\}$. The edges were illustrated as red dashed arrow and black dashed arrow. Red dashed arrow means the retrieval route and black dashed arrow means the empty cargo movement route between Distribution Centers (DCs). From this figure, there were also 24 vehicles used for retrieval and seven of them were moving with empty cargos to another DC through route G until L before they go back to manufacturer with the retrieved items. The route G, H, I, K, and L were used by one vehicle for each route and the route J was used by two vehicles. Route A to G were used as retrieved routes with 10 vehicles went through route A, two vehicles went through route B, three vehicles went through route C, seven vehicles went through route D, and one vehicles each went through route E and F. For example in the blue rectangle areas at Fig. 6, vehicle number 4 directly went back to manufacturer, vehicle number 2 went to DC 9, and vehicle number 14 and 18 went to DC 12. Previously, these vehicles were transporting products to DC 1. That means some vehicles moving to another DC site to retrieved RGB while the other retrieved the RGB from the same DC site of distribution. The vertices that were not connected on the graph depicted in Fig. 6 means that there were no RGB retrieved from these vertices. This was happening because it was located quite far away from the vertices that serve as destination node for distribution. Therefore, the model did not choose these vertices as origin node for RGB retrieval.

The model determined the route by selecting the solutions, which has lower fitness value for the first objective (minimizing transportation cost). Each solution has sub-chromosome 5 and 9, which were used to encode the DCs. These DCs were subsequently set the distribution and retrieval's route. Therefore, the solution which has a better (in this case a lower one) fitness value for the first objective would steadily survive until the last generation. This consideration for computational purpose is much different compared with classical minimum spanning tree approach such as Kruskal's, Prim's, and Boruvka's algorithm. For instance, Kruskal's algorithm determines the route by firstly sorting the weight or length of the edges in an increasing order. For each edge e in a sorted order, if the endpoints

of e are disconnected in the graph, then it added the lowest edge e to the graph, repeated the process until all the vertices on the graph were connected without the edges forming a cycle. The problem on this research required that the vehicles were going back to the manufacturer with RGB loaded, therefore forming a non-tree network. Because of that, a Kruskal's algorithm and the other classical minimum spanning tree approaches were not in accordance to solve the problem on this research. In order to evaluate the solution, a best practice manual result called as referenced value, set as denominator of value found by the model. The solution's similarity was formulated as the average distance of each variables between two solutions. The notation o_{ref} refers to the solution used as reference and the notation n refers to the number of variable(s) considered to calculate the solution's similarity. It defined that

G_a^o as value of variable a in solution o

$G_a^{o_{ref}}$ as referenced value of variable a in solution o

Also, the equation for solution's similarity is:

$$Sim_n^o = \frac{\sum_{a=1}^n \left(\frac{G_a^o}{G_a^{o_{ref}}} \right)}{n} \quad (12)$$

A few samples of the solutions that were generated according to section 6 were plotted in a pairwise diagram as provided in Fig. 7. The difference between each solution may occur and the solutions may be similar or not. For this purpose, the solutions were selected randomly as it is almost an infinite combinatorial case to compare the whole solutions. The solution numbers 3, 19, 89, 229, and 396 were randomly chosen as the sample. From Fig. 7, the comparison between variables in sub-chromosomes of distribution (P1-P4) with sub-chromosome of DC destination showed that each variable in the solutions was graphically inclined to a specific point. Similarly, comparison between sub-chromosomes of retrieval (R1-R3) with sub-chromosome of DC origin also showed that each variable in the solutions was graphically incline to a specific point. However, the other comparison showed that each variable of the solutions was graphically spread in the variable space. This result indicates that the solutions has a similar value for each variable in sub-chromosome of DC destination and sub-chromosome of DC origin. This diagram showed that the characteristics of the solutions from the model was similar in selecting the Distribution Centers as destination or origin nodes but quite differ in determining the amount of distribution and retrieval. The single reference similarity according to equation 12 was used to calculate the similarity value between solutions. For instance, the solution number 3 was used as a reference.

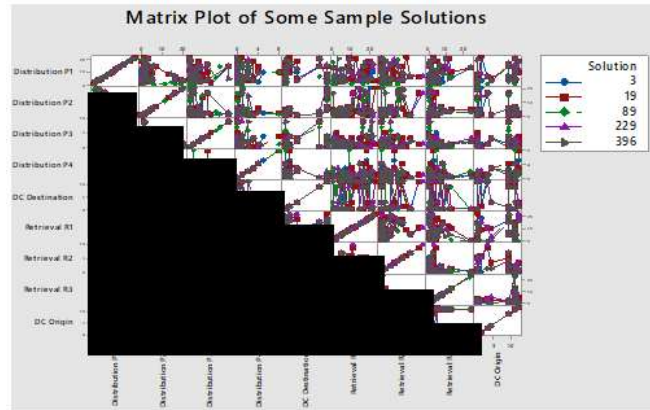


Fig. 7. Pairwise comparison between solution 3, 19, 89, 229, and 396

Using the range of similarity between 0.9-1.1, only solution number 229 has an overall similarity inside the range. The details of the overall similarity of the sample is provided in Table 5.

Table 5

Overall similarity value on the single reference

| Solution number | Similarity |
|-----------------|------------|
| 19 | 0.896 |
| 229 | 1.000 |
| 396 | 1.158 |
| 89 | 1.414 |

According to this table, Solution number 229 was similar with solution number 3. That means, the route and freight load (amount of distribution and retrieval) is similar with solution number 3 that has been presented before.

5.3 Verification and validation

Verification for the model was considered on each mathematical model of goals and constraints as explained before. The verification process compares the results from the model with a manual mathematical calculation using the help of MS. Excel. Input for this Excel file was derived from an example of mentioned solution 3 above. According to these values, the verification results proved that there was no difference between model calculations with mathematical calculations. For instance, the model calculates the first objective and the second objective equal to Rp 5,620,372 and 1.999, respectively. The manual calculation also produces the same values. Moreover, the manual calculation of constraints' value proved that it never violated the threshold provided in the form of parameters of case study. Validation based on operational graphic for framework validation and extreme condition test for result validation had been performed. In the operational graphic validation, the Pareto front at generation 500,000th; 1,000,000th; and 1,600,000th were observed. The results showed that for each increase of generation, the solutions were getting more optimum. Objective function of Eq. (1) and Eq. (2) were deployed to compare the optimality of the solutions between generations. The next validation stage was with extreme condition test. The extreme case was set with the products demand and RGBs requirement equal to zero. It is assumed that in this case, the model should distribute and retrieve nothing. The result showed that the model distributes zero amount of product and retrieve zero amount of retrieved item for this extreme case. The solution has the value for the first objective equal to zero and the second objective equal to zero/zero. This implies that the solution did not transport anything. The validation result proves that the utilization of framework and the generated results were valid.

6. Conclusion

In this work, the model to optimize freight scheduling in an IRLP on high to intermediate network level (manufacturer to Distribution Center) was proposed. The proposed model was formulated to satisfy two conflicting objectives while satisfying nine constraints. It utilized a meta-Non-dominated Sorting Genetic Algorithm-II to solve the problem. The problem that was presented as a case study was successfully optimized. This result was interpreted in terms of amount of distribution and route selection of transportation. A further interpretation on solutions similarity provided a larger insight for comparison between solutions.

6.1 Advantages and disadvantages

From the results and discussion above, the model proved to yield best solutions with regard to transportation at minimum cost and order responsiveness, while satisfying both forward logistics

constraints as well as reverse logistic constraints. Unfortunately, the model only covers freight schedules in a single manufacturing site and single stop transportation problems.

6.2 Managerial impacts

The implementation of the proposed model to the case study is promisingly enhance the manufacturing's logistics system performance. The implementation of the model is clearly increasing the order responsiveness while minimizing the transportation cost. However, the management needs to change the policy in order to conform to the model's assumption, such as the DC site and amount of retrieval is not necessarily the same with DC site and amount of distribution. Moreover, the calculation of previous RGBs requirement would change, that is by forecasting the demand 4 days to come. This also implies that there is no need for the manufacturing to maintain the stock of RGBs within shorter periods. In order to implement the model, the management needs to provide the daily data for this proposed model input. The management also needs to have a firm understanding of the most desired value of responsiveness as well as the ratio between forward logistics responsiveness and reverse logistics responsiveness

6.3 Future work

For future works, since the proposed model runs only for the case of single manufacturing site with a single stop transportation case, it is necessary to accommodate problems with multi manufacturing sites and multi stops transportation.

Acknowledgement

The authors would like to thank the anonymous referees for constructive comments on earlier version of this paper.

References

- Agrawal, S., Singh, R., Murtaza, Q. (2016). Disposition decisions in reverse logistics: Graph theory and matrix approach. *Journal of Cleaner Production*, 137, 93–104.
- Barwaldt, M., Franzin, R., Casarin, V., dos Santos, A. (2014). Using the theory of graphs on the implementation of bike lane in small towns, in: *Procedia of Social and Behavioral Sciences*. Presented at the XVIII Congreso Panamericano de Ingeniería de Tránsito, Transporte y Logística, Elsevier, Santander (ES), pp. 350–358.
- Baumik, P.K. (2015). Supply chain network design based on integration of forward and reverse logistics. *Global Business Review*, 16(4), pp. 680-699.
- Cataruzza, D., Absi, N., Feillet, D., Vidal, T. (2013). A memetic algorithm for the multi trip vehicle routing problem. *European Journal of Operational Research*, 236, 833–848.
- de Brito, M., Dekker, R. (2003). A framework for reverse logistics (ERIM Report), Research in Management. Erasmus Research Institute Of Management, Rotterdam (NL).
- de Oliveira, L. S. D., & Saramago, S. F. (2010). Multiobjective optimization techniques applied to engineering problems. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 32(1), 94-105.
- Deb, K., & Agrawal, R.B. (1994). Simulated binary crossover for continuous search space (Technical Report No. IITK/ME/SMD-94027). Indian Institute of Technology, Kanpur.
- Deb, K., Goyal, M. (1996). A combined genetic adaptive search (GeneAS) for engineering design. *Computer Science and Informatics*, 26, 30–45.
- Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6, 182–197.

- Démare, T., Bertelle, C., Dutot, A., Lévêque, L. (2017). Modeling logistic systems with an agent-based model and dynamic graphs. *Journal of Transport Geography*, 62, 51–65.
- Diestel, R. (2005). *Graph Theory*. Springer, New York (US).
- Dondo, R.G., Mendez, C.A. (2016). Operational planning of forward and reverse logistic activities on multi-echelon supply-chain networks. *Computers and Chemical Engineering*, 88, 170–184.
- Fazlollahtabar, H. (2018). Reverse supply chain vehicle routing problem: similarity pattern model in Supply chain management models : forward, reverse, uncertain, and intelligent foundations with case studies book, pp. 217-225.
- Fleischmann, M., Beullens, P., BLOEMHOF-RUWAARD, J. M., & Van Wassenhove, L. N. (2001). The impact of product recovery on logistics network design. *Production and Operations Management*, 10(2), 156-173.
- Gudehus, T., Kotzab, H. (2009). *Comprehensive Logistic*. Springer, Berlin (DE).
- Guidice, F. (2013). Graph-based approach for modeling, simulation, and optimization of life cycle resource flows. *Reverse Supply Chain book*, ed, Gupta, S.M. CRC Press.
- Hadka, D. (2011). *Beginner's Guide to the MOEA Framework*.
- Hadka D. (2017). MOEA Framework version 2.12. Available at: <https://github.com/MOEAFramework/MOEAFramework/releases/download/v2.12/MOEAFramework-2.12-Source.tar.gz>
- Kacprzyk, J., & Pedrycz, W. (Eds.). (2015). *Springer Handbook of Computational Intelligence*. Springer, Berlin (DE).
- Khajavi, L.T., Seyed-Hosseini, S.M., Makui, A. (2011). An integrated forward/reverse logistics network optimization model for multi-stage capacitated supply chain. *Scientific Research*, 3, 229–235.
- Lee, D.H., & Dong, M. (2008). A heuristic approach to logistics network design for end-of-lease computer products recovery. *Transportation Research Part E*, 44, 455–474.
- Likaj, R., Shala, A., Mehmetaj, M., Hyseni, P., Bajrami, X. (2013). Application of graph theory to find optimal paths for the transportation problem. Presented at the 15th Workshop on International Stability, Technology, and Culture, The International Federation of Automatic Control, Prishtina (KO).
- Parkhi, S., Jagadeesh, D., Kumar, R.A. (2014). A study on transportation cost optimization in retail distribution. *Journal of Supply Chain Management System*, 3, 31–38.
- Pishvae, M.S., Jolai, F., Razmi, J. (2009). A stochastic optimization model for integrated forward/reverse logistics network design. *Journal of Manufacturing System*, 28, 107–114.
- Price, E., & Ostfeld, A. (2014). Optimal Water System Operation Using Graph Theory Algorithms, in: *Procedia Engineering*. Presented at the 16th Conference on Water Distribution System Analysis, Elsevier, pp. 502–208.
- Rao, R., & Savsani, V. (2012). *Mechanical Design Optimization Using Advanced Optimization Techniques*. Springer, Berlin (DE).
- Rogers, D., & Tibben-Lembke, R. (1999). *Going Backwards: Reverse Logistics Trends and Practices*. Pittsburgh (PA). Reverse Logistics Executive Council, Pittsburgh (PA).
- Siregar, H.H. (2012). Solving Vehicle Routing Problem on the Distributions of Highland Vegetables Using Genetic Algorithm (Case study PT Saung Mirwan) (Undergraduate Thesis). Bogor Agricultural University, Bogor (ID).
- Tseng, Y., Yue, W.L., Taylor, M.A.P. (2005). The role of transportation in logistics chain, in: *Proceedings of the Eastern Asia Society for Transportation Studies*. Presented at the The 6th EASTS conference, Bangkok (TH), pp. 1657–1672.
- Zaki, S.A., Mousa, A.A.A., Genedi, H.M., Elmekawy, A.Y. (2012). Efficient multiobjective genetic algorithm for solving transportation, assignment, and transshipment problems. *Applied Mathematics*, 3, 92–99.
- [Oracle]. 2017. MySQL Community Edition version 5.7.18. Available at: <https://dev.mysql.com/downloads/mysql/>



© 2019 by the authors; licensee Growing Science, Canada. This is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).