

Inventory Routing Problem for Perishable Product with Dynamic Demand and Spoilage Rate

Krisna Wahyudi^{1a}, I Gede Agus Widyadana^{2b}, I Nyoman Sutapa^{3c}

^{1,2,3}Master of Industrial Engineering Departement, Petra Christian University, Jalan Siwalankerto 121-131, Surabaya 60236, Indonesia
^akrisnawahyudi17@gmail.com, ^bgede@petra.ac.id, ^cmantapa@petra.ac.id

Abstract. The paper presents an inventory routing problem for perishable products using dynamic demand and spoilage rate data. This problem has three main costs for the simulation criteria: holding inventory cost, routing shipment cost, and perishable product cost with a single supplier and many retailers. Instead of the problem, there are 2 set decision variables inventory level range and shipping route. The simulation uses a Genetic Algorithm method to find the minimum cost. It shows that the dynamic data demand can be optimized, and the dynamic demand data have bigger cost consequences than the static demand data.

Keywords: Inventory Routing Problem, Perishable Product, Optimization, Dynamic Demand, Dynamic Spoilage Rate, Genetic Algorithm.

1. Introduction

Vendor-Managed Inventory (VMI) is inventory management controlled by the supplier for their retailer. In the VMI system, suppliers manage retailers' inventory levels from shipping quantity and shipping route in one planning period [1]. Inventory Routing Problem (IRP) is one of the VMI methods to optimize the cost of inventory routing. Inventory routing optimization is an important thing for fulfilling the demand market with optimal cost. IRP is a basic of optimizing inventory levels from retailers for fulfilling the demand market with minimum transport and inventory from the system [2]. The IRP object is to minimize the cost of shipping and inventory from the planning horizon without causing retail's inventory to run out of stock.

The perishable products give more challenges in inventory management and also attempting to minimize defective unsold products. The perishable products have a random lifetime caused by the environment, such as temperature and humidity, and also uncertainty of transport time and product harvest condition [3]. The random lifetime of the perishable product causes inventory cost because of that defective unsold product [4-6].

The previous paper developed the IRP model to optimize the routing problem of perishable products with a static demand and spoilage rate. In this paper, the IRP model for perishable products decides inventory stock level and shipping route with a dynamic demand and spoilage rate. This model is solved by the Genetic Algorithm (GA) method to get optimal

inventory stock level and shipping route with optimal cost. This proposed model was tested by random dynamic demand and spoilage rate data. The problem result of this proposed model can be used to decide the inventory level of each retail and shipping route with optimal cost and fulfill the dynamic demand.

2. Literature Review

2.1. Inventory Routing Problem (IRP)

IRP is a primary method for optimizing the inventory level of retailers to fulfill market demand with a minimum shipping and inventory cost from a system [2]. IRP is one of the VMI solutions that suppliers manage retail's inventory level. In the previous paper, IRP was proposed for many problem methods such as perishable product, maritime, inbound, multi-product, stochastic periodic, green, and cyclic with two versions of IRP deterministic and stochastic [2].

From previous IRP papers, there is two main cost problem from IRP, inventory holding cost and shipping cost [1, 4, 6, 7]. Holding cost is an expense for storing products at each retail warehouse. That holding cost usually counts by the amount of inventory stock in the warehouse. Shipping cost is an expense of shipping transport from supplier to each retail and back to supplier. These costs have several parameters such as the amount of holding cost of each product from each retail, retail distance, the inventory level of each period, shipping capacity, retail holding capacity, and others. There is previous related literature with each part that can be seen in Table 1.

Table 1. Previous IRP paper

Reference	Solution Method	Perishable Products	Transshipment	Single Supplier	Single Carrier	Dynamic Demand & Spoilage Rate	Dynamic Location of Distribution Center
This paper	Genetic Algorithm	V		V	V	V	
[4]	Genetic Algorithm + Taguchi Approach	V	V	V	V		
[5]	Decomposition procedure	V			V		
[7]	Genetic Algorithm	V		V			
[8]	Myopic Algorithm			V			V

2.2. Perishable Product

A perishable product is a product that has a random lifetime. The product that ships from supplier to retail may have a different lifetime [3]. The perishable product has an uncertain lifetime that is challenged to fulfill market demand with the amount of products spoilage that can't be sold. Examples of perishable products are packaged food, food material, fruits, vegetables, and others.

In IRP model [4] said that perishable products affect the cost from the product's price when can't sell the defected product. The spoilage rate of perishable products is directly proportional to the amount of inventory stock level. In the IRP model, the spoilage rate variable is a function to count the amount of defective product from each retail in each period.

2.3. Genetic Algorithm

A genetic Algorithm is a biologically motivated approach which popular in solving a complex optimization problem. GA makes sets of solutions called the population to get the best solution. GA has an operator such as crossover, mutation, and selection are utilized to get new solution sets (offspring) to reach the desired solution [4]. There are three important things in GA, chromosome representation, fitness value function, and population regeneration. Chromosome representation is a set structure of decision variables to solve the problem. The fitness value function is a formulation value which is the benchmark value of each set chromosome. Population regeneration is a process to get a new set of chromosomes (offspring). GA is commonly used to solve a non-linear problem, such as in [4] and [7]. Both of those papers discuss about IRP for perishable products and using the GA method to solve the problem.

3. Problem Description

In this section, we propose a mathematical model of inventory routing problem for a perishable product with dynamic demand and spoilage rate which the supplier should decide an optimal solution of retail's inventory level and shipping route. This mathematical model is based on [4] with the adaptation of dynamic demand and spoilage rate. These model objects are single product, single supplier, single shipping vehicle, and multi-period with multi retail. The objective of this problem is to minimize the total cost of the system and fulfill the entire demand in each period.

3.1. Notation

3.1.1. Parameters

Notation	Description
h_i	Holding cost of retail i
T	Number of periods
M	Number of retail
$d_i(t)$	Demand of retail I in t period
$Sr_i(t)$	Spoilage rate product of retail I in t period
$Sp_i(t)$	Number of defect product of retail I in t period
Q	Shipping capacity in one period
c_{ij}	Shipping distance cost from retail i to retail j
C_i	Holding capacity of retail i
P	Product's price
$I_i(0)$	Initial stock of retail i

3.1.2. Decision variables

Notation	Description
L_i	Minimum stock level retail i
M_i	Maximum stock level retail j
$X_{ij}(t)$	Binary Decision, 1 if pass a route from retail i to retail j , and else is 0
$q_i(t)$	Quantity product sent to retail i in t period

3.2. Mathematical Model

The objective function of this model is to minimize the cost of the inventory system. There are three costs from this model, holding inventory cost represented in the first term, defect product cost in the second term, and shipping route cost in the third term that as shown in equation (1)

$$\text{Min } Z = \sum_{t=1}^T \sum_{i=1}^M h_i I_i(t) + P \sum_{t=1}^T \sum_{i=1}^M Sp_i(t) + \sum_{t=1}^T \sum_{i=1}^M \sum_{j=i+1}^M c_{ij} X_{ij}(t) \quad (1)$$

Subject to:

$$I_i(t) = I_i(t-1) - Sp_i(t-1) + q_i(t) - d_i(t) \quad i = 1..M; t = 1..T \quad (2)$$

$$Sp_i(t) = Sr_i(t) \times I_i(t) \quad i = 1..M; t = 1..T \quad (3)$$

$$I_i(t) \leq M_i \leq C_i \quad i = 1..M; t = 1..T \quad (4)$$

$$I_i(t) \geq L_i \quad i = 1..M; t = 1..T \quad (5)$$

$$\sum_{i=1}^M q_i(t) \leq Q \quad t = 1..T \quad (6)$$

$$\sum_{i=0}^M X_{ij}(t) = \sum_{i=0}^M X_{ji}(t) \quad j = 1..M; t = 1..T \quad (7)$$

$$X_{ij}(t) \in \{0,1\} \quad i = 1..M; j = 1..M; i \neq j; t = 1..T \quad (8)$$

$$Sr_i(t) \in \{0\%, \dots, 100\% \} \quad i = 1..M; t = 1..T \quad (9)$$

$$T, M \in \{0, 1, 2, 3, \dots\} \quad (10)$$

$$h_i, d_i(t), Q, C_i, I_i(t), L_i, M_i, q_i(t) \geq 0 \quad (11)$$

Number of products shipped at t period equals to fulfill retail's stock until maximum stock level (M_i) if retail's last stock ($t-1$ period) is less than minimum stock level (L_i). The number of products shipped can be seen in equation (12).

$$q_i(t) = M_i - (I_i(t-1) - Sp_i(t-1) - d_i(t))i = 1..M; t = 1..T \quad (12)$$

Equation (2) shows a retailer's stock level at t period equals to last period stock level and shipped product number reduced by last period spoilage product and demand product at t period. Equation (3) is a number of spoilage products at t period equals to spoilage rate at t period multiplied by the stock level at t period. Equation (4) is a constraint of stock level, which must be less than maximum stock level and also maximum retail's stock capacity. Equation (5) is a constraint of stock level, which must be greater or equal to the minimum stock level. Equation (6) is a constraint of total shipping, which must be less or equal to the shipping capacity of each period. Equation (7) is a constraint of the shipped route. Equation (8) is a constraint of shipping route is a binary. Equation (9) is a constraint of spoilage rate product. Equation (10) is a constraint where the number of retails and periods are integer. Equation (11) is a non-negativity constraint.

4. Genetic Algorithm

4.1. Dynamic Data Simulation

A model with dynamic data was proposed by [8] through dynamic distribution center simulation. In this paper, the dynamic simulation was adopted to generate data set of dynamic demand and spoilage rate. This part generates demand and spoilage rate data randomly for each retail at each period and simulates for each data set with the solution using the Genetic Algorithm method to find an optimal solution for all data set. The representation of demand and spoilage rate data can be seen in Figure 1.

		Retail Number				
		1	2	3	4	5
Period	1	55	56	24	42	37
	2	33	43	20	41	53
	3	52	23	34	40	46
	4	44	38	20	23	29
	5	44	32	33	35	39

		Retail Number				
		1	2	3	4	5
Period	1	11	15	12	7	12
	2	5	9	5	9	7
	3	8	11	8	5	7
	4	13	12	9	14	6
	5	10	8	14	6	15

Figure 1. Representation of demand and spoilage rate data

4.2. Chromosome Representation

Chromosome representation is a representation of decision variables from the problem model. The decision variable of this model is inventory stock level and shipping route. Inventory stock level generates a range of stock amount for each retail i with L_i for minimum stock and M_i for maximum stock with a random number from 0 to maximum retail i capacity. Inventory stock level representation can be seen in Figure 2.

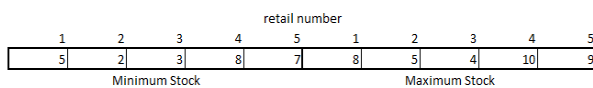


Figure 2. Inventory Stock Level Representation

The shipping route is an order of delivery from the supplier to each retail and returns to the supplier in each period. If the quantity of delivery for retail i is zero, then the route of retail i will be zero. This route representation is transferred into $X_{ij}(t)$ with binary value. Shipping route representation can be seen in Figure 3.

		Retail Number				
		1	2	3	4	5
period	1	0	1	0	3	2
	2	1	0	2	3	0
	3	2	0	0	1	3
	4	0	0	1	2	3
	5	4	2	5	3	1

Figure 3. Shipping Route Representation

Each chromosome has a solution of inventory stock level and shipping route. Each of these set representations has a fitness value from this objective function model. The smaller fitness value is the optimal solution.

4.3. Population

A population is a group of chromosomes that are generated randomly. The chromosome is generated randomly by the system for P times in the first iteration. In the next iteration, the population of chromosomes is generated from the Genetic Algorithm operator, such as crossover, mutation, and elitism. For each iteration, the system randomly pairs two chromosomes using the roulette wheel method into crossover until P offspring is generated.

4.4. Crossover

Crossover is one of the GA operators that generate new offspring chromosomes from pairs of parent chromosomes. In this process, these chromosomes change parts of the parent chromosomes to each other and generate new offspring chromosomes by using the one-point crossover. There are 2 parts of each chromosome, inventory stock level and shipping route. Inventory stock level change the half minimum and maximum part with the other chromosome that can be seen in Figure 4.

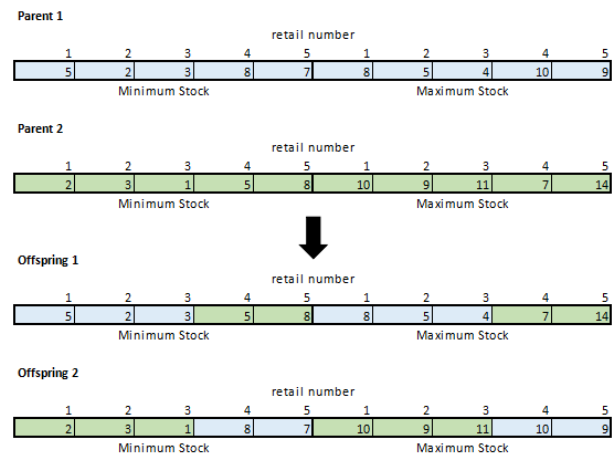


Figure 4. Inventory Stock level Crossover

Shipping route chromosome also changes half their parts with other chromosomes. The crossover of this shipping route chromosome can be seen in Figure 5.

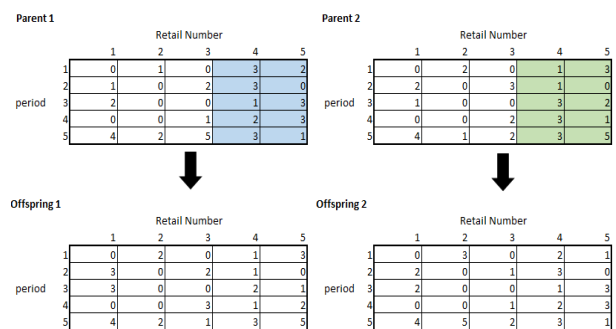


Figure 5. Shipping Route Crossover

4.5. Mutation

The mutation is a GA operator that changes the part of each new offspring chromosome randomly. In this process, each new offspring chromosome mutates each part with the mutation rate (*Mr*) parameter as a random mutation decision. The value of *Mr* parameter in this simulation is set to 0.6 for inventory stock level mutation. Each part of the chromosome generates a random number and changes to a new random number with interval 0 to maximum retail capacity, and a constraint for the minimum must be smaller than maximum, and the maximum must be bigger than the minimum. If the random number is bigger than *Mr*. Mutation of inventory stock level can be seen in Figure 6.

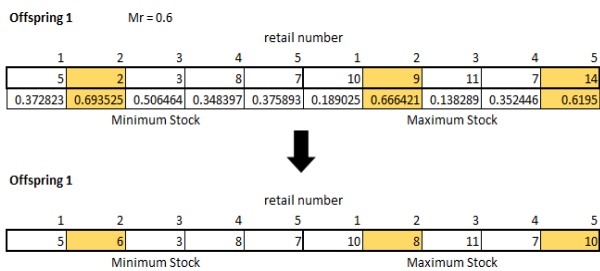


Figure 6. Inventory Stock Level Mutation

The shipping route also generates the random number on each chromosome's part and mutates two parts in each period that have a bigger random number. Shipping route mutation can be seen in Figure 7.

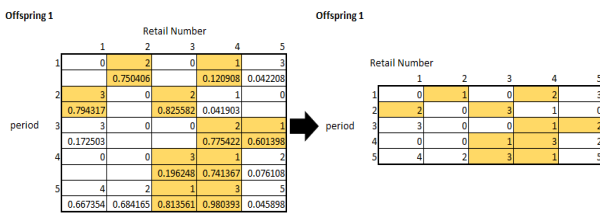


Figure 7. Shipping Route Mutation

4.6. Elitism

Elitism is the last GA operator that preserves several parent chromosomes which have the lowest fitness value with the specified amount. The preserved parent chromosomes combined with new offspring chromosomes population for next iteration. The purpose of this elitism process is to preserve a good parent chromosome that has a good fitness value. This proposed model uses a fixed number of elitism.

4.7. Stopping criteria

Stopping criteria is a condition statement to end GA iteration. This proposed model uses a fixed stopping criteria with a number of iterations parameter (*g*).

5. Numerical Example

In this section, the proposed Genetic Algorithm method and representation are run to solve three sets of data simulation samples. These are represented through every five sets of demand and spoilage rate. This simulation focused on simulating different demand numbers. Each of the data sets of demand generated randomly with uniform distribution normal interval dynamic demand, normal interval static demand, and small interval dynamic demand in [20 - 60] units, [20 - 60] units, and [30 - 40] units, respectively with same spoilage rate [5 - 15]%. This simulation has five periods with ten retails and 750 units supplier's shipping capacity for each period. The main problem of this simulation sample is the dynamic demand and spoilage rate. This simulation shows the decision variable should be fulfilled to all dynamic demand and spoilage rate samples. The cost of this simulation was generated randomly with a uniform distribution interval for holding cost [10 - 25], retail cost distance [20 - 50], and fixed 40 for the product price. The other parameters were holding cost each retail and number of elitism in [60 - 100] and ten respectively. The population and iteration numbers are a combination of population 80, 100, 120, and 140, and also iteration 100, 150, 200, and 250. The cost of this simulation is represented by each fitness value solution. Fitness value for each solution/

Table 2. Simulation Result

Number	Iteration	Population	Process Result					
			Dynamic Demand 20-60		Static Demand 20-60		Dynamic Demand 30-40	
			Best Fitness	Execution Time (s)	Best Fitness	Execution Time (s)	Best Fitness	Execution Time (s)
1	100	80	46301	787	41708.4	764	34129.2	903
2	100	100	46844.8	920	42791.6	947	33772.6	1095
3	100	120	45929.6	1170	43828.2	1150	33781.2	1249
4	100	140	45753	1302	41591	1359	33659	1455
5	150	80	46958.4	1080	42491.6	1153	33537.6	1242
6	150	100	46233.2	1440	42653.6	1407	34110.8	2272
7	150	120	46333.8	1707	42435	1692	33296	2149
8	150	140	46605.4	2017	41782.2	1937	33871.4	2568
9	200	80	46362.4	1527	41707.6	1497	33808.4	1728
10	200	100	45907.8	1888	42598.6	1824	33437	2084
11	200	120	46033.8	2309	41732.8	2243	33562.6	2265
12	200	140	46474.6	2953	41461	3064	33408.6	2629
13	250	80	45551.6	1945	41430.8	1891	33311.6	2134
14	250	100	45230.2	2324	41327.2	2451	33252.8	2441
15	250	120	45123	2814	41738	2859	33094.8	2895
16	250	140	44978.2	3311	41297.2	3281	32568	3587

chromosome obtained from the average of each cost demand and spoilage rate data set simulation. This Genetic Algorithm is implemented by using PHP HTML programming language and database storage using MySQL to facilitate subsequent development using real demand and spoilage rate data and also take advantage of the flexibility access web technology.

6. Result & Discussion

The simulation has one supplier, ten distributors, and five periods. The supplier controlled the inventory levels and shipping routes for each distributor. The simulation use combination of iteration number in 100, 150, 200, and 250 and population number in 80, 100, 120, and 140 with three different data dynamic demand 20-60, static demand 20-60, and dynamic demand 30-40. The result of each simulation shows that it has a cost reduction with the best fitness value at 250 iterations and 140 population and has a more execution time in each added iteration or population. The simulation combination and simulation result can be seen in Table 2.

Instead of the best fitness value with more iteration and population, the execution time is also increased. The optimal execution time with the optimal result is in 250 iterations and 140 populations which have a significant cost reduction of 1.53% and 0.68%, respectively. The increment time and cost reduction comparison can be seen in Table 3.

Table 3. Increment Time and Cost Reduction Comparison

Iteration Change	Avg. Increment Time (%)	Avg. Cost Reduction (%)
100 to 150	56.44	-0.01
150 to 200	28.59	0.75
200 to 250	23.72	1.53

Population Change	Avg. Increment Time (%)	Avg. Cost Reduction (%)
80 to 100	27.50	-0.18
100 to 120	17.20	0.27
120 to 140	19.58	0.68

The first result is a normal interval dynamic demand [20 - 60] units iteration with 19.8% average cost reduction and optimal fitness value 44978.2, inventory level decision retail 1 to 10 in [48 - 66], [40 - 64], [56 - 57], [60 - 66], [2 - 61], [36 - 62], [14 - 63], [59 - 68], [4 - 72], [57 - 63], respectively, and shipping route decision period 1 to 5 in [0-2-4-6-10-9-3-7-1-8-5-0], [0-5-8-3-2-1-7-10-4-6-9-0], [0-2-4-5-10-7-9-6-8-3-1-0], [0-7-8-6-3-1-2-9-4-10-5-0], [0-4-6-5-3-2-8-1-9-7-10-0], respectively. This result has cost reduction of initial solution from 56357.6 to 44978.2 shown in Figure 8.

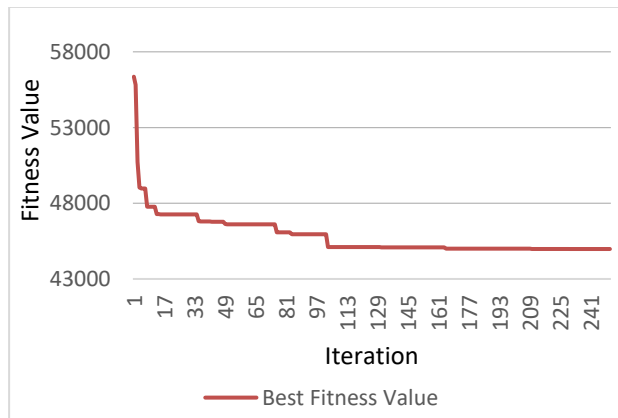


Figure 8. Cost Reduction of Normal Interval Dynamic Demand

The second result is a normal interval static demand [20 - 60] units iteration with 20.3% average cost reduction and optimal fitness value 41297.2, inventory level decision retail 1 to 10 in [59 - 66], [46 - 83], [53 - 64], [21 - 60], [53 - 57], [7 - 58], [39 - 63], [0 - 59], [58 - 68], [68 - 85], respectively, and shipping route decision period 1 to 5 in [0-1-7-8-4-5-3-9-2-10-6-0], [0-8-10-6-3-4-1-7-5-9-2-0], [0-9-1-7-5-2-10-4-3-6-8-0], [0-10-8-7-2-6-5-9-4-1-3-0], [0-3-9-10-1-8-5-2-7-6-4-0], respectively. This result has cost reduction of initial solution from 50763.2 to 41297.2 shown in Figure 9.

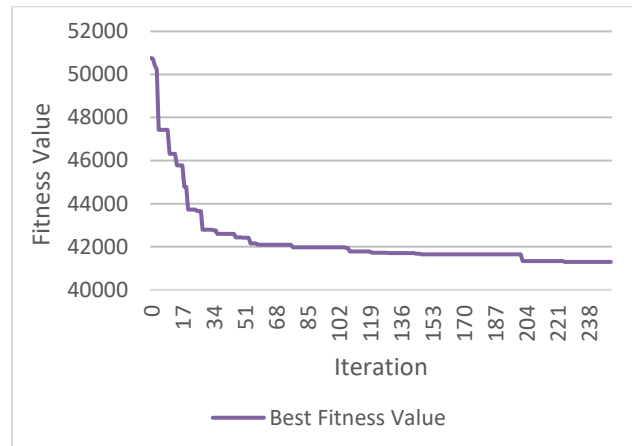


Figure 9. Cost Reduction of Normal Interval Static Demand

The third result is a small interval dynamic demand [30 - 40] units iteration with 24.2% average cost reduction and optimal fitness value 32568, inventory level decision retail 1 to 10 in [14 - 46], [41 - 52], [41 - 46], [21 - 40], [46 - 46], [25 - 50], [20 - 45], [0 - 49], [37 - 47], [7 - 84], respectively, and shipping route decision period 1 to 5 in [0-3-2-4-10-9-5-7-8-6-1-0], [0-5-10-3-7-1-8-2-9-4-6-0], [0-4-9-6-5-10-2-3-7-8-1-0], [0-5-3-7-6-2-10-4-8-9-1-0], [0-2-3-8-9-4-10-6-5-1-7-0], respectively. This result has 34.1% cost reduction of initial solution from 44669.8 to 32568 shown in Figure 10.

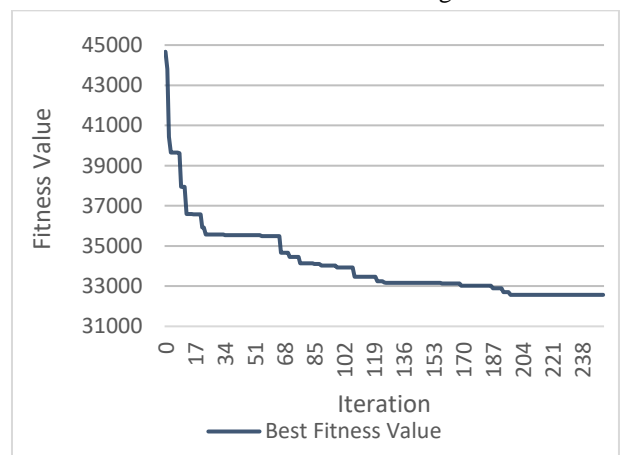


Figure 10. Cost Reduction of Small Interval Dynamic Demand

The model in this paper combines the IRP model of perishable products with dynamic data of demand and spoilage rate from retails on each period. There are samples of dynamic data through the previous study about dynamic distribution center simulation, which converted into dynamic data of demand and spoilage simulation and generated set data

simulation of previous IRP model. There are three main costs in this model, holding inventory cost, defect product cost and shipping route cost. The solution should be fulfilling each demand and spoilage rate simulation and have the best cost. By using Genetic Algorithm, the result of this simulation shows that the dynamic demand can be optimized by the result's pattern with fixed stopping criteria iteration and have a higher cost than static demand. The best result of this simulation is small interval dynamic demand which has 24.2% cost reduction, followed by normal interval dynamic and static demand which have 19.8% and 20.3% cost reduction. This result shows that smaller interval has a better reduction and dynamic and static demand have the same reduction that shows the dynamic can be solved as a static problem. But, the optimal best solution shows that dynamic demand data has bigger optimal cost consequences than static demand data. Instead of the best solution, the execution time shows that the most optimal cost reduction is in 250 iterations and 140 population. In this paper, the IRP model has a single product, single supplier, and single-carrier object. Therefore, the dynamic demand and spoilage rate can be optimized by this IRP model. So, the model can be improved with other inventory routing problem objects or constraints such as multiple products, multiple suppliers, multiple carriers, or anything else.

7. Conclusions

The simulation has one supplier, ten distributors, and five periods. The supplier controlled the inventory levels and shipping routes for each distributor. The simulation use combination of iteration number in 100, 150, 200, and 250 and population number in 80, 100, 120, and 140 with three different data dynamic demand 20-60, static demand 20-60, and dynamic demand 30-40. The result of each simulation shows that it has a cost reduction with the best fitness value at 250 iterations and 140 population and has a more execution time in each added iteration or population.

References

- [1]. Guimarães, T., Coelho, L., Schenekemberg, C., & Scarpin, C. (2019). The two-echelon multi-depot inventory-routing problem. *Computers & Operations Research*, *101*, 220-233. doi:10.1016/j.cor.2018.07.024
- [2]. Yadollahi, E., Aghezzaf, E.-H., Walraevens, J., Raa, B., & Claeys, D. (2019). Evaluating approximate solution models for the stochastic periodic inventory routing problem. *Journal of Manufacturing Systems*, *50*, 25-35. doi:10.1016/j.jmsy.2018.11.001
- [3]. Ketzenberg, M., Gaukler, G., & Salin, V. (2018). Expiration dates and order quantities for perishables. *European Journal of Operational Research*, *266*(2), 569-584. doi:10.1016/j.ejor.2017.10.005
- [4]. Azadeh, A., Elahi, S., Farahani, M. H., & Nasirian, B. (2017). A genetic algorithm-Taguchi based approach to inventory routing problem of a single perishable product with transshipment. *Computers & Industrial Engineering*, *104*, 124-133. doi:10.1016/j.cie.2016.12.019
- [5]. Hu, W., Toriello, A., & Dessouky, M. (2018). Integrated inventory routing and freight consolidation for perishable goods. *European Journal of Operational Research*, *271*(2), 548-560. doi:10.1016/j.ejor.2018.05.034
- [6]. Rohmer, S. U. K., Claassen, G. D. H., & Laporte, G. (2019). A two-echelon inventory routing problem for perishable products. *Computers & Operations Research*, *107*, 156-172. doi:10.1016/j.cor.2019.03.015
- [7]. Hiassat, A., Diabat, A., & Rahwan, I. (2017). A genetic algorithm approach for location-inventory-routing problem with perishable products. *Journal of Manufacturing Systems*, *42*, 93-103. doi:10.1016/j.jmsy.2016.10.004
- [8]. Segura, E., Carmona-Benitez, R. B., & Lozano, A. (2014). Dynamic Location of Distribution Centres, a Real Case Study. *Transportation Research Procedia*, *3*, 547-554. doi:10.1016/j.trpro.2014.10.010