AN INVENTORY ROUTING PROBLEM FOR DETERIORATING ITEMS WITH DYNAMIC DEMAND AND SPOILAGE RATE

I Gede Agus WIDYADANA
*Master of Industrial Engineering Departement, Petra Christian University,
Surabaya, Indonesia*
gede@petra.ac.id

Krisna WAHYUDI
*Master of Industrial Engineering Departement, Petra Christian University,
Surabaya, Indonesia*
krisnawahyudi17@gmail.com

I Nyoman SUTAPA
*Master of Industrial Engineering Departement, Petra Christian University,
Surabaya, Indonesia*
mantapa@petra.ac.id

Abstract: Inventory routing problems (IRP) are among important tools to be used for implementing vendor manage inventory. Many researchers try to develop methods for solving inventory routing problem, however, only a few developed methods for inventory routing problems for spoilage items. In reality, many items are deteriorated and spoiled during transportation and storage period. In this paper, we developed a model and methods to solve the inventory routing problem for deteriorating items with dynamic demand and spoilage rate, i.e., demand varies and items spoil during planning periods. Those cases are more realistic since many commodities such as fruits and vegetables have dynamic demand and spoilage rate. A Genetic Algorithm and Particle Swarm Optimization are developed to solve the problem with various demands in a specific planning period since the problem is Np-hard. A numerical example and sensitivity analysis are conducted to verify the model, and to get management insight it. The result is interesting and support general hypothesis that dynamic demands result in higher inventory cost than the static demands, and the increasing demand results in increasing inventory cost. Also, the results show that increasing demand and deteriorating rates significantly affect

the total cost, therefore, the developed model is important and significantly useful to be used for solving IRP with dynamic demand and spoilage items.

Keywords: Inventory, Spoilage, Dynamic Demand, Genetic Algorithm, Particle Swarm Optimization.

MSC: 65K10.

1. INTRODUCTION

Inventory Routing Problem deals with inventory and distribution issues since both costs are dominant costs in the supply chain, ([4] and [12]). Therefore, many organizations try to streamline stock holding and transportation expenses. The IRP is an optimization inventory problem considering transportation costs so to minimize transportation and inventory cost [16]. More than 30 years, researchers have payed their attention to the IRP and gave various models and solutions, see [3]. The IRP model that includes IRP with the continuous move was introduced by [14]. Aghezzaf and Landeghem [1] developed IRP with multi-tours, and then stochastic demand IRP was introduced by [6], and Liu et.al. [9] developed an IRP model with time windows.

Recently, many IRP models consider perishable items, which have a random lifetime regarding environmental conditions, e.g., temperature and humidity, the uncertainty of transportation time, and harvest time [8]. Hu et.al. [7] developed a simple IRP for perishable items using decomposition and local search methods. Decomposition and local search are efficient methods, however, to get a more effective solution, metaheuristic methods can be applied. A Genetic Algorithm is used by [2] to solve an IRP problem with perishable items and single vendor. They extended IRP model by setting transshipment from one store to another stores. A similar model, and amethod, for perishable items was developed by [5], however, they did not consider transshipment between stores. Widyadana and Irohara [15] extended IRP models for perishable items by considering deteriorated items and items delivered under specific time windows. In the model, items are not only deteriorated during storing period in warehouses but are also, deteriorating during the transportation period. The model is extended by [11], where they considered multi-objectives by regarding environmental issues instead of only one objective.

In this model, we further developed an IRP great model from [2] and introduced a new model for solving more realistic conditions. We use dynamic demand instead of static demand, and a dynamic spoilage rate, since it is difficult to predict spoilage rate for agriculture commodities such as vegetables, meat, and fruits. We developed a model that is more realistic to be applied for agriculture items. As the problem is an NP-hard, therefore Genetic algorithm and Particle Swarm Optimization are used to solve it. Genetic Algorithm is an effective method used for solving IRP problems ([2] and [5]). The other effective metaheuristic method used to solve NP-hard problems is Particle Swarm Optimization ([14] and [10]).

2. MODEL DEVELOPMENT

We developed the Inventory routing model, using [2], with dynamic demand and spoilage rate. Some assumptions of this model are:

- Demand rate distribution during the replenishment period (T) is known.

- Deteriorating rate during replenishment period (T) is known Yellow

In this model, we assume that there is one product, one vendor, and one delivery player within a multi-period. Notations are:

Parameters:

h_i	: Inventory cost at store I ($)
T	: Replenishment period
M	: Numbers of store
$d_i(t)$	: Delivery volume from store i at period t
$Sr_i(t)$	: Deteriorating rate in store i at period t
$Sp_i(t$	: Number of reject in store i at period t
Q	: Delivery capacity (unit)
c_{ij}	: Delivery cost from store i to store j ($)
C_i	: warehouse capacity at store i
Pr	: Product price ($)
$I_i(0)$	: Initial stock in store i

Variables

L_i	: Minimum stock in store i
H_i	: Maximum stock in store i
$X_{ij}(t)$	: $\begin{cases} 1, \text{ if product delivered from store } i \text{ to store } j \\ 0, \text{ otherwise} \end{cases}$

$$MinZ = \sum_{t=1}^{T}\sum_{i=1}^{M} h_i I_i(t) + Pr\sum_{t=1}^{T}\sum_{i=1}^{M} Sp_i(t) + \sum_{t=1}^{T}\sum_{i=1}^{M}\sum_{j=i+1}^{M} c_{ij} X_{ij}(t) \quad (1)$$

Constraints:

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

$$Sp_i\left(t\right) = Sr_i\left(t\right)I_i\left(t\right) \qquad i = 1..M; t = 1..T \tag{3}$$

$$I_i\left(t\right) \leq C_i \qquad i = 1..M; t = 1..T \tag{4}$$

$$I_i\left(t\right) \geq L_i \qquad i = 1..M; t = 1..T \tag{5}$$

$$\sum_{i=1}^{M} q_i\left(t\right) \leq Q \qquad t = 1..T \tag{6}$$

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

$$X_{ij}\left(t\right) \in \{0,1\} \qquad i = 1..M; j = 1..M; i \neq j; t = 1..T \tag{9}$$

$$Sr_i\left(t\right) \in \{0\%, \ldots, 100\%\} \qquad i = 1..M; t = 1..T \tag{10}$$

$$T, M \in \{0, 1, 2, 3, \ldots\} \tag{11}$$

$$h_i, d_i\left(t\right), Q, C_i, I_i\left(t\right), L_i, M_i, q_i\left(t\right) \geq 0 \tag{12}$$

The objective function is shown in Eq. 1, where the total cost consists of inventory cost, reject cost, and delivery cost. Equations 2 to 12 show the model constraints. Stock quantity in period t depends on the previous period stock quantity, number of rejected stock, incoming and outgoing product (Eq. 2). Eq. 3 shows that the rejected quantity is equal to rejected rate times quantity of stock. Maximum stock quantity cannot overcome store capacity (eq. 4). Stock quantity in every store in every period must be bigger than the minimum stock (5). Delivery quantity of each vendor is equal to capacity delivery to all stores (6). Whenever there is delivery from i to j, the vehicle will outbound from j (7). Eq. 8 shows delivery quantity when the number of stock is less than the maximum stock.

2.1. Genetic Algorithm

Genetic algorithm used in this paper is a simple GA with:

1. Chromosome representation

retail number

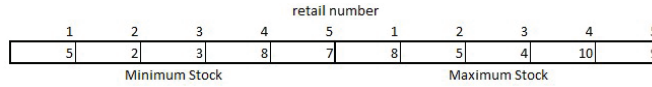|	1	2	3	4	5	1	2	3	4	5
	5	2	3	8	7	8	5	4	10	9

Minimum Stock   Maximum Stock

Figure 1: Minimum and maximum stock chromosome

Chromosome represents three variables, i.e., minimum stock, maximum stock, and routing for each period. Minimum and maximum stocks are represented using integer values as shown in Figure 1. Notation for the minimum stock is L_i and the maximum stock is M_i.

The third variable is delivery routing for each period. A chromosome is represented by integer values that show the sequence of the store visited by the vehicle. When one store is not visited, the chromosome is represented by zero (0). A chromosome for vehicle routing can be seen in Figure 2.

Retail Number

period		1	2	3	4	5
	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 2: Chromosome representation for vehicle routing

2. Fitness function and selection

The GA fitness function is derived from equation 1, and parents selection is using roulette wheel rule where chromosome with lower fitness value has a higher possibility to be chosen as a parent.

3. *Crossover*

The *Crossover* method that is used in this paper is a one-point crossover.

4. *Mutation*

The Mutation process is done inside of one chromosome where each cell has 40% possibility to be mutated. When a cell is chosen to be mutated, it changes randomly between a specific minimum and maximum values.

5. *Elitism*

Elitism is a process to keep the best chromosome in each generation automatically, it become an offspring in the next generation.

6. *Stopping criteria*

Stopping criteria sets the GA method to be stopped. In this paper stopping criteria is a predetermined number of generations.

2.2. Particle Swarm Optimization

Particle Swarm Optimization algorithm works as particles move together to follow the positions that have been found so far. Some important things that have to be set up to get good PSO are particle representation, position, velocity, fitness function, global best, and local best.

1. Particle representation

   Particle representation for minimum and maximum stock is similar to chromosome representation in GA, however, vehicle routing representation is different. PSO uses float values instead of the sequence as shown in GA. Particle representation for vehicle routing is shown in Figure 3. The sequence of the vehicle will follow the particle from the lowest to the highest values, where value 0 shows that the retail is not visited at that period. For example, the route for the vehicle in period one in Figure 3 is from the vendor to stores 4-2-5 and the return to vendor.
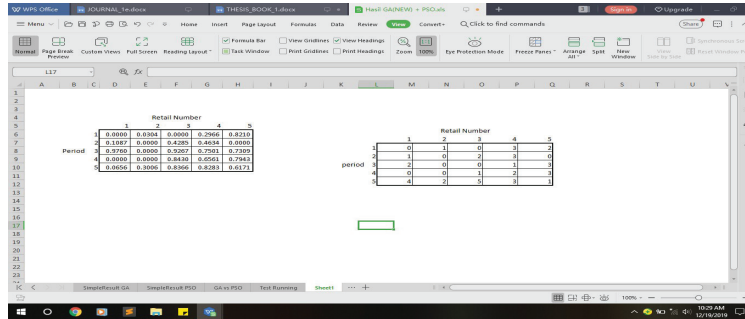

Figure 3: Particle representation of vehicle routing

2. Particle velocity

   Particle velocity will be updated at every iteration to move every particle closer to the best particle in local and global positions. The velocity for each particle can be updated using equation 12.

$$v_i^{k+1} = wv_i^k + c_1 rand_1 \times \left(pbest_i - p_i^k\right) + c_2 rand_2 \times \left(gbest - p_i^k\right) \quad (12)$$

Where the gbest is the best particle position for all previous iterations, and pbest is the position of a particle from the previous iteration.

3. Particle position

The later position of each particle is moved with its velocity as shown in equation 13.Particle position

Later position of each particle is moved with its own velocity as shown in equation 13.

$$p_i^{k+1} = p_i^k + v_i^{k+1} \tag{13}$$

4. Fitness value

The fitness value is similar to the GA fitness value, where the fitness value is derived from equation 1.

5. The best solution and stopping criteria

The best solution is derived form the best solution from the last iteration, where the number of iterations is determined in advanced.

3. A NUMERICAL EXAMPLE AND RESULT

The numerical example uses the main data from data generation. There are two types of demand; the first case is dynamic demand and the second case is static demand. Demand data are generated uniformly from 20-60. The other data such as distance, inventory cost, warehouse capacity at each store, initial stock, delivery capacity, and product price are generated consecutively; 10, 5, uniform [20 - 50], uniform [10 - 25], uniform [60 - 100], uniform [60 - 100], 750, and 40.

We set varies population sizes and the number of iterations to get the best parameter for the Genetic Algorithm for both cases. The best fitness function for case 2 is shown in Figure 4.

Figure 4: The best fitness value for case 2 with 250 Iterations and 140 population

Figure 4 shows that the GA is not trapped to local optimal easily. The two cases show that, at least, the solutions convergent at the 200th generation. We set the number of generations equal to 250 to guarantee that the solution has been convergent and the best solution is 18.65% less than the initial fitness value. Therefore, the GA results in a significant improvement compared to the initial solution.

For the PSO there are 32 combinations of the population (P), iteration (G), and cases. The best solution for case 1 is derived from 250 iterations and 140 population. The best fitness values for case 1 is shown in Figure 5. The solutions show that the longest iteration for getting the best fitness values convergent is 198 iterations, therefore we set the number of iterations to be 250 to guarantee that the best fitness value has been convergent. The best fitness value of PSO is 32.14% better than the initial solution.

Figure 5: The PSO best fitness value for case 1 with 250 Iterations and 140 population

The best solution for case 1 and case 2 for GA and PSO are shown in Table 1.

The Table shows that the PSO outperforms GA in terms of the best fitness value and computation time, therefore, we use PSO to get management insight of static and dynamic demands.

Table 1: Table 1. The optimal values and running time of GA and PSO

1*	Optimal Cost ($)		Running Time (S)	
—-	GA	PSO	GA	PSO
—— Case 1	45123	44296.6	3311	2850
—— Case 2	41297.2	41058.6	3281	2659
——				

3.1. Table 2 shows a comparison between case 1 with dynamic demand and case 2 with static demand. The Table shows that dynamic has less inventory cost but higher spoilage cost and delivery cost However, the spoilage and delivery costs are not significant. Case 1 has a higher cost than case 2 since stores have to keep more stocks to prevent lost sales. The dynamic demand results in a 9.1% higher inventory cost than the static demand. Therefore, it is important for organizations to keep demand as stable as possible, in each period, to reduce inventory cost.

Table 2: Table 2. Fitness values for case 1 and case 2

Case 1			Case 2		
Holding cost	$	35362.6	Holding cost	$	32421.4
Spoilage Cost	$	8032.0	Spoilage Cost	$	8096.0
Transportation Cost	$	737.8	Transportation Cost	$	795.0
Penalty Cost	$	0.0	Penalty Cost	$	0.0
Total cost	$	**44132.4**	**Total cost**	$	**41312.4**

A sensitivity analysis is used to evaluate the effect of spoilage rate to the total

cost. We use spoilage rate 5% , 7% , and 9% and keep all parameters at the same values. The sensitivity analysis result is shown in Table 3.

Table 3: Table 3. Fitness values for varies of spoilage rate

Spoilage Rate 5%		Spoilage Rate 7%		Spoilage Rate 9%	
Holding cost	$ 23382.4	Holding cost	$ 24292.0	Holding cost	$ 25069.6
Spoilage Cost	$ 3840.0	Spoilage Cost	$ 4824.0	Spoilage Cost	$ 5376.0
Transportation Cost	$ 808.0	Transportation Cost	$ 902.0	Transportation Cost	$ 699.0
Total cost	$ 28030.4	Total cost	$ 30028.0	Total cost	$ 31144.6

Table 3 shows an increasing 2% spoilage rate form 5% to 7% , resulting in increasing holding costs from 3% to 4% , and increasing the total cost from 3.7% to 7.2%. It shows that the spoilage rate significantly affect holding cost and spoilage cost. Therefore, this model is suitable to be used for items with high spoilage cost such as vegetables, fruits, and meat. For the set parameter values determined in this numerical example, the holding cost is the highest cost for the fitness values. This model shows that spoilage rate has high contribution to the total cost, therefore, management has to keep spoilage rate as low as possible. Tools or machines for keeping low spoilage rate and variation such as refrigerator, refrigeration truck, vacuum packaging, and drying machines are important to be applied for all business related with spoilage items. This finding is useful to support decisions and strategies for some business in countries like Indonesia, where managements resist to invest in machines and technologies. They think that total inventory cost for spoilage items are not too significant compared to investment cost.

4. CONCLUSION

Inventory routing problem model is important to be used in the vendor manage inventory scheme, where the vendor delivers and keeps his stock to some retail stores. In this paper, an IRP for spoilage items is developed since spoilage items are common in reality. Instead of considering static demand, this paper considers dynamic demand. The model is an NP-hard, therefore Genetic Algorithm and Particle Swarm Optimization are developed to solve it. Both methods are effective, and a numerical example is used to compare their performances.The results show that the PSO outperforms GA in terms of solution quality and computation time, and the total cost for dynamic demand is 9% higher than the total cost of static demand. A sensitivity analysis is conducted to analyst the effect of spoilage rate to the total cost. It shows that the spoilage rate significantly affects the total cost. The numerical example and sensitivity analysis show that this model should be

used to handle spoilage items and dynamic demand. Increasing of spoilage rate significantly affects total inventory cost, therefore it is important for business to invest in some tools or machines to keep low spoilage rate and variations. Savings on inventory cost could be higher than machines investment cost on plausible time. This paper assumes that there is no transshipment between stores, therefore, the next research can consider transshipment since it is commonly used by some retail stores to reduce costs. The other future research by considering environmental issues for delivering items.

REFERENCES

[1] Aghezzaf E-H., Raa B., & Landeghem, H.H., "Modeling inventory routing problems in supply chains of high consumption products", *European Journal of Operational Research*, 169, (2006) 1048-1063.

[2] Azadeh, A., Elahi, S., Farahani, M. H., & Nasirian, B., "A genetic algorithm-Taguchi based approach to inventory routing problem of a single perishable product with transshipment", *Computers & Industrial Engineering*, 104 (2017) 124-133.

[3] Coelho L.C., Cordeau J.F., & Laporte G., "Thirty years of inventory routing", *Transportation Science*, 48 (1) (2014) 1-19.

[4] Guimarães, T., Coelho, L., Schenekemberg, C., & Scarpin, C., "The two-echelon multi-depot inventory-routing problem", *Computers & Operations Research*, 101 (2019) 220-233.

[5] Hiassat, A., Diabat, A., & Rahwan, I., "A genetic algorithm approach for location-inventory-routing problem with perishable products", *Journal of Manufacturing Systems*, 42 (2017) 93-103.

[6] Huang S-H., & Lin P-C., "A modified ant colony optimization algorithm for multi-item inventory routing problems with demand uncertainty", *Transportation Research Part E,* 46 (2010) 598-611.

[7] Hu, W., Toriello, A., & Dessouky, M., "Integrated inventory routing and freight consolidation for perishable goods", *European Journal of Operational Research* 271 (2) (2018) 548-560.

[8] Ketzenberg, M., Gaukler, G., & Salin, V., "Expiration dates and order quantities for perishables", *European Journal of Operational Research*, 266 (2) (2018) 569-584.

[9] Liu S-C., & Lee W-T., "A heuristic method for the inventory routing problem with time windows", *Expert Systems with Applications*, 38 (2011) 13223-13231.

[10] Norouzi, N., Sadegh-Amalnick, M., & Alinaghiyan, M., "Evaluating of the particle swarm optimization in a periodic vehicle routing problem", *Measurement*, 62 (2015) 162-169.

[11] Rau, H., Budiman S.D., and Widyadana, G.A., "Optimization of the multi-objective gree cyclical inventory routing problem using discrete multi-awarm PSO method", *Transportation Research Part E:Logistics and Transportation Review*, 120, 51-75.

[12] Rohmer, S. U. K., Claassen, G. D. H., & Laporte, G., "A two-echelon inventory routing problem for perishable products", *Computers & Operations Research*, 107 (2019) 156-172.

[13] Sadeghi, J., Sadeghi, S., & Niaki, S. T. A., "Optimizing a hybrid vendor-managed inventory and transportation problem with fuzzy demand: An improved particle swarm optimization algorithm", *Information Sciences*, 272 (2014) 126-144.

[14] Savelsbergh, M., and Song J. H., "Inventory routing with continuous moves", *Computers & Operations Research*, 34 (2007) 1744-1763.

[15] Widyadana, G.A., and Irohara T., "Modelling multi-tour inventory problem for deteriorating items with time windows", *Scienta Iranica*, 26 (2) (2019) 932-941.

[16] Yadollahi, E., Aghezzaf, E.-H., Walraevens, J., Raa, B., & Claeys, D., "Evaluating approximate solution models for the stochastic periodic inventory routing problem", *Journal of Manufacturing Systems*, 50 (2019) 25-35.