An Integrated Model for Lot Sizing with Supplier Selection Considering Quantity Discounts, Expiry Dates, and Budget Availability

Teguh E.N. Sitepu[#1], Andi Cakravastia [*2]

[#]Supply Chain Management Department, Institut Teknologi Harapan Bangsa
Jl. Dipati Ukur 80-84, Bandung, Indonesia
[1]teguh_sitepu@ithb.ac.id

[*]Industrial Engineering Department, Institut Teknologi Bandung
Jl. Ganesa 10, Bandung, Indonesia
[2]andi@mail.ti.itb.ac.id

***Abstract*— In this paper, a dynamic multi-product multi-period lot sizing with supplier selection problem (DLSSP) with quantity discount, expiry dates, and budget availability is presented. Demand of products for each period are independent and known. The cost consists of ordering, purchasing, transportation, expiry, holding, and interest charge. The objective is to find the optimal order quantity of all items in each period to minimize inventory cost. A mixed integer nonlinear model programming (MINLP) is first developed to model the problem. Since model is hard to solve using exact method, Genetic Algorithm (GA) and Simulated Annealing (SA) is applied, in which design parameters are set using Taguchi method. Computational results demonstrate the applicability of the proposed model and comparing the results show efficiency of both algorithms as well. The results show that, while both algorithms have statistically similar performances, proposed SA is the better algorithm in all problems.**

***Keywords*—** *lot size, supplier selection, quantity discount, perishable product, simulated annealing*

1. Introduction

Procurement is the process of obtaining goods and services within a supply chain. Managers must structure procurement with a goal of increasing supply chain surplus [1]. Ref. [2] claimed that purchased goods and services are one of the largest elements of cost for many firms. Determination of right procurement strategy can increase a firm's profit [3].

Dynamic lot sizing problem (DLSP) is one of the basic procurement strategy, introduced by Wagner & Within [4]. They considered only one item and one supplier. Ref. [5] developed DLSP model into dynamic multi-product multi-period lot sizing model with supplier selection problem (DLSSP). Since then, many developed models of DLSSP had been done by considering condition in real life. This paper considered quantity discount, expiry dates, and budget availability.

Quantity discount is an effective strategy for many suppliers to promote their product. On the other hand, buyer can buy product with cheaper price when buy product over certain amount. DLSSP model considered quantity discount had been done by [6], [7], and [8]. Their models considered unlimited lifetime item.

In real life, some products have limited lifetime. For example: food, pharmaceutical, and fashion product. Limited lifetime product will decay if it's not used until expired. So, inventory pick-up policy is important for limited lifetime product [9]. Ref. [10] and [11] had been developed DLSP model considered limited lifetime product.

Based on literature review, majority of developed DLSSP models considered unlimited budget. In real life, a firm has limited budget, so cash is the heart of all businesses [12]. If procurement cost less than budget availability, the firm will buy with cash. But if procurement cost greater than budget availability, the firm must buy with credit and pay interest charge to the bank.

In this paper, DLSSP model for limited lifetime product developed by considering quantity discount and budget availability. Ref. [13] stated this problem is NP-hard problem. So, this paper proposed MINLP model to solve problem and comparing exact method with metaheuristic algorithm. Genetic Algorithm (GA) and Simulated Annealing (SA) presented.

The remaining of this paper is organized as follows. In section 2, some of related researches are reviewed. In section 3, problem is described. In

International Journal of Supply Chain Management
IJSCM, ISSN: 2050-7399 (Online), 2051-3771 (Print)

section 4, the formulation of proposed model. In section 5, construction of metaheuristic algorithms described. In section 6, experiments are carried out. In section 7, conclusion is presented and future directions discussed.

2. Related Works

Lot size problem can be categorized by it's characteristic. One of the category is lot size problem with discrete time, deterministic demand, and finite planning horizon. This category called dynamic lot sizing problem (DLSP) and first introduced by Wagner & Within [4]. Basic DLSP model assumed product can buy only from one supplier. In real life, some products aren't monopolized so the firm can select supplier whose offer cheaper price. That condition made DLSP model developed into dynamic lot sizing with supplier selection problem (DLSSP). Ref. [5] was first introduced DLSSP considering multi-product. Ref. [14] developed DLSSP model considering limited supplier capacity. Proposed model is mixed integer nonlinear programming (MINLP) and solved using LINGO software. Ref. [13] developed new method to solve Ref [5] model. Ref. [13] presented Reduce and Optimize Approach (ROA).

In procurement policy, supplier can offer discount quantity based on purchasing amount. This strategy can make supplier keep leading in price competitiveness. On the other hand, buyer can buy product in certain quantity to take advantage from supplier offering. Ref. [6] and [15] developed DLSP model considering quantity discount and limited warehouse capacity. Ref. [6] implemented GA and Memetic Algorithm (MA). While Ref. [15] developed heuristic algorithm based on GA. Ref. [16] developed GA to solve DLSPP model considering quantity discount. Proposed model has objective function to minimize ordering cost, purchasing cost, transportation cost, and holding cost. Ref. [8] added third dimension to Fordyce Webster Algorithm (FWA) to solved DLSSP model.

Based on literature review, majority of lot sizing models considered unlimited budget. Ref. [17] developed probabilistic inventory model considering budget availability and limited warehouse capacity. Proposed model was nonlinear and solved using LINGO software. Ref. [18] developed DLSP model considering budget availability and quantity discount. Solution procedure was using GA and PSO. Furthermore, Ref. [19] developed Ref. [18] into bi-objective mixed integer nonlinear programming (MINLP). Proposed model has objective function to minimize total inventory cost and storage area. Ref. [12] developed DLSP model to maximize profit. The firm must pay interest-bearing loan if budget availability not enough to buy product with cash. On the other hand, the firm will gain interest-bearing deposit if budget availability extant at one period.

All developed DLSP models above considering unlimited lifetime product. In real life case, some products have limited lifetime, so it will decay if it's not used until expired. Ref. [10] developed dynamic lot sizing and scheduling problem considering limited lifetime product. Consideration of limited lifetime part and product increased planning complexity. Ref. [11] developed DLSP considering pick-up policy by First-Expired-First-Out (FEFO). Proposed model was mixed integer linear programming (MILP) through linearization and then solved by GA and PSO.

3. Problem Description

We assume a retailer who sell multi products to customer. The retailer buys products from multi suppliers who offer different ordering, purchasing, and transportation cost. For purchasing cost, every supplier offers different all-unit discount policy shown in Fig 1. Every period, the retailer has limited budget to buy products. If budget isn't enough to buy with cash, the retailer must borrow some money from bank so interest charge cost increasing.

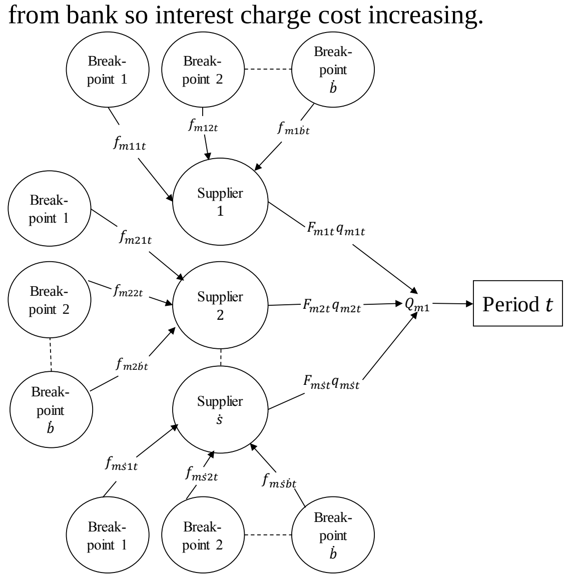

Figure 1. Procurement system of the proposed model

At beginning every period, the retailer will receive demand from customer. Next, the retailer must select supplier and set order quantity to every supplier. Proposed inventory model based on Ref. [11] and shown in Fig 2. Unused product will keep to the next period and the lifetime will decrease one period. For every stored product, retailer must pay holding cost to warehouse maintenance. While for every expired product, the firm must pay expiry cost to recycle or disposal activity. The retailer must optimize procurement policy to minimize total inventory cost along planning horizon. Total inventory cost consists of ordering, purchasing, transportation, expiry, holding, and interest charge.

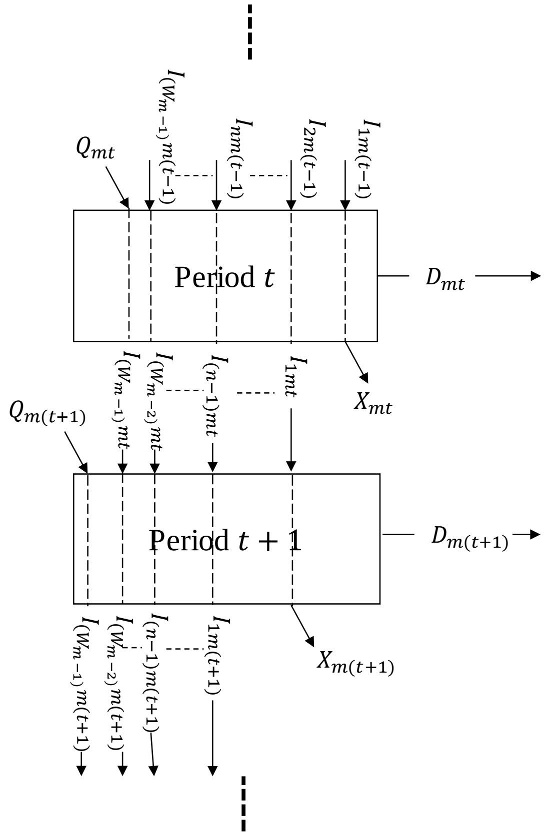

Figure 2. Inventory system of the proposed model

4. Mathematical Model

4.1 Assumptions

Assumptions in this problem are defined as follows:

a. Demand for each period is independent and known.
b. Initial and ending inventory are zero.
c. Each product has the same space at vehicle.
d. Shortage is not allowed.
e. Lead time is known and delivered at once in the beginning of a period.
f. First Expired-First Out (FEFO) policy is considered for picking up inventory.
g. The capacity of supplier and warehouse is unlimited.
h. The price is dependent on the order quantity.
i. Unused budget cannot be carried over to subsequent periods
j. Each product has a finite lifetime that greater than planning period

4.2 Notations

4.2.1 Indices

Indices in this problem are defined as follows:

m products
n remaining lifetime
s suppliers
b breakpoint
t periods

4.2.2 Parameters

Parameters in this problem are defined as follows:

BUD_{mt} budget availability for product type *m* in period *t*
β_t interest rate in period *t*
W_m lifetime of product type *m*
H_m holding cost of product type *m*, per unit per period
j_m expiry cost of product type *m*, per unit per period
C_{ms} ordering cost of product type *m* from supplier *s*
K_s transportation cost per vehicle from supplier *s*
D_{mt} demand of product type *m* in period *t*
L_{msb} the lower bound quantity of product type *m* from supplier *s* with price break *b*
U_{msb} the upper bound quantity of product type *m* from supplier *s* with price break *b*
P_{msb} unit purchase cost of product type *m* from supplier *s* with price break *b*
N the maximum vehicle batch size
V a large number

4.2.3 Decision Variables

Decision variables in this problem are defined as follows:

X_{mt} expired quantity of product *m* in period *t*
I_{nmt} inventory level of product type *m* with *n* periods of lifetime remaining in the end of period *t*
R_{st} number of vehicle from supplier *s* in period *t*
Q_{mt} total purchase quantity of product type *m* in period *t*
$P(q_{mst})$ purchase cost for a unit product type *m* based on the discount of supplier *s* in period *t*
α_{mt} the maximum number of product type *m* bought with cash in period *t*
γ_{mt} number of product type *m* that bought with credit in period *t*
q_{mst} purchase quantity of product type *m* from supplier *s* in period *t*
CA_{mt} method of payment product type *m* in period *t*

Y_{mt} a binary variable, set equal to 1 if a purchase is made, and 0 if no purchase is made, of product type *m* in period *t*

F_{mst} a binary variable, set equal to 1 if a purchase is made, and 0 if no purchase is made, of product type *m* from supplier *s* in period *t*

f_{msbt} a binary variable, set equal to 1 if a purchase is made, and 0 if no purchase is made, of product type *m* from supplier *s* with price break *b* in period *t*

4.3 Developed Mathematical Model

Due to the budget availability, one of the following conditions is satisfied for each period:

Condition 1 : $Q_{mt} = 0 \ \forall\, m, t$ (1)

Condition 2 : $0 < Q_{mt} \leq \alpha_{mt} \ \forall\, m, t$ (2)

Condition 3 : $Q_{mt} > \alpha_{mt} \ \forall\, m, t$ (3)

Expression (1) means that no product type *m* purchased in period *t*. Expression (2) means that product type *m* in period *t* (Q_{mt}) purchased with cash because the quantity is equal or less than the maximum number of product type *m* can bought with cash in period *t* (α_{mt}). Expression (3) means that product type *m* in period *t* (Q_{mt}) purchased with credit because the quantity is greater than α_{mt}.

The nonlinear budget availability constraint sets are converted to linear expressions with the help of a new set of binary variables ($CA_{\sigma mt}$).

$$\mathrm{CA}_{\sigma mt}\begin{cases}1, \text{if condition } \sigma \text{ is sustained} \\ 0, \text{otherwise}\end{cases} \quad \forall\, \sigma, m, t \quad (4)$$

This way, the mathematical model of the problem is stated as follows:

$$\min \sum_{t\in T}\left[\begin{array}{c}\sum_{s\in S}\left(\begin{array}{c}(F_{mst}C_{ms}) \\ \sum_{m\in M} +(P(q_{mst})F_{mst}q_{mst}) \\ + R_{st}K_s\end{array}\right) \\ +\sum_{m\in M}\left(\begin{array}{c}X_{mt}j_m + \sum_{n=1}^{W_m} I_{nmt}H_m + \\ \gamma_{mt}(1+\beta_t)\sum_{s\in S}P(q_{mst})\,F_{mst}\end{array}\right)\end{array}\right] \quad (5)$$

subject to

$$I_{nm0} = I_{nmt'} = 0, \quad \forall\, n, m \quad (6)$$

$$Q_{mt} = \sum_{s\in S} q_{mst}, \quad \forall\, m, t \quad (7)$$

$$R_{st} \geq \frac{\sum_{m\in M} q_{mst}}{N}, \quad \forall\, s, t \quad (8)$$

$$\begin{cases} D_{mt} \leq I_{1m(t-1)} + V.(1 - z_{1mt}) \\ X_{mt} \geq I_{1m(t-1)} - D_{mt} - V.(1 - z_{1mt}) \\ X_{mt} \leq I_{1m(t-1)} - D_{mt} + V.(1 - z_{1mt}) \\ I_{1mt} \geq I_{2m(t-1)} - V.(1 - z_{1mt}) \\ I_{1mt} \leq I_{2m(t-1)} + V.(1 - z_{1mt}) \\ \ldots \\ I_{nmt} \geq I_{(n+1)m(t-1)} - V.(1 - z_{1mt}) \\ I_{nmt} \leq I_{(n+1)m(t-1)} + V.(1 - z_{1mt}) \\ \ldots \\ I_{(W_m-1)mt} \geq Q_{mt} - V.(1 - z_{1mt}) \\ I_{(W_m-1)mt} \leq Q_{mt} + V.(1 - z_{1mt}) \end{cases} \quad \forall m, t \quad (9)$$

$$\begin{cases} I_{1m(t-1)} - V.(1 - z_{2mt}) < D_{mt} \\ I_{1m(t-1)} + I_{2m(t-1)} + V.(1 - z_{2mt}) \geq D_{mt} \\ X_{mt} \geq -V.(1 - z_{2mt}) \\ X_{mt} \leq V.(1 - z_{2mt}) \\ I_{1mt} \geq I_{1m(t-1)} + I_{2m(t-1)} - D_{mt} - V.(1 - z_{2mt} \\ I_{1mt} \leq I_{1m(t-1)} + I_{2m(t-1)} - D_{mt} + V.(1 - z_{2mt}) \\ \ldots \\ I_{nmt} \geq I_{(n+1)m(t-1)} - V.(1 - z_{2mt}) \\ I_{nmt} \leq I_{(n+1)m(t-1)} + V.(1 - z_{2mt}) \\ \ldots \\ I_{(W_m-1)mt} \geq Q_{mt} - V.(1 - z_{2mt}) \\ I_{(W_m-1)mt} \leq Q_{mt} + V.(1 - z_{2mt}) \end{cases} \forall m, t \ (10)$$

$$\begin{cases} I_{1m(t-1)} + I_{2m(t-1)} - V.(1 - z_{3mt}) < D_{mt} \\ I_{1m(t-1)} + I_{2m(t-1)} + I_{3m(t-1)} + V.(1 - z_{3mt}) \geq D_{mt} \\ X_{mt} + I_{1mt} \geq -V.(1 - z_{3mt}) \\ X_{mt} + I_{1mt} \leq V.(1 - z_{3mt}) \\ I_{2mt} \geq I_{1m(t-1)} + I_{2m(t-1)} + I_{3m(t-1)} - D_{mt} - V.(1 - z_{3mt}) \\ I_{2mt} \leq I_{1m(t-1)} + I_{2m(t-1)} + I_{3m(t-1)} - D_{mt} + V.(1 - z_{3mt}) \\ I_{3mt} \geq I_{4m(t-1)} - V.(1 - z_{3mt}) \\ I_{3mt} \leq I_{4m(t-1)} + V.(1 - z_{3mt}) \\ \ldots \\ I_{nmt} \geq I_{(n+1)m(t-1)} - V.(1 - z_{3mt}) \\ I_{nmt} \leq I_{(n+1)m(t-1)} + V.(1 - z_{3mt}) \\ \ldots \\ I_{(W_m-1)mt} \geq Q_{mt} - V.(1 - z_{3mt}) \\ I_{(W_m-1)mt} \leq Q_{mt} + V.(1 - z_{3mt}) \end{cases} \quad \forall m, t \quad (11)$$

. .

$$\begin{cases} \sum_{n=1}^{W_m-2} I_{nm(t-1)} - V.(1 - z_{(W_m-1)mt}) < D_{mt} \\ \sum_{n=1}^{W_m-1} I_{nm(t-1)} + V.(1 - z_{(W_m-1)mt}) \geq D_{mt} \\ X_{mt} + \sum_{n=1}^{W_m-3} I_{nmt} \geq -V.(1 - z_{(W_m-1)mt}) \\ X_{mt} + \sum_{n=1}^{W_m-3} I_{nmt} \leq V.(1 - z_{(W_m-1)mt}) \\ I_{(W_m-2)mt} \geq \sum_{n=1}^{W_m-1} I_{nm(t-1)} - D_{mt} - V.(1 - z_{(W_m-1)mt}) \\ I_{(W_m-2)mt} \leq \sum_{n=1}^{W_m-1} I_{nm(t-1)} - D_{mt} + V.(1 - z_{(W_m-1)mt}) \end{cases} \quad \forall m, t \quad (12)$$

$$\begin{cases} \sum_{n=1}^{W_m-1} I_{nm(t-1)} - V.(1 - z_{W_m mt}) < D_{mt} \\ Q_{mt} + \sum_{n=1}^{W_m-1} I_{nm(t-1)} + V.(1 - z_{W_m mt}) \geq D_{mt} \\ X_{mt} + \sum_{n=1}^{W_m-2} I_{nmt} \geq -V.(1 - z_{W_m mt}) \\ X_{mt} + \sum_{n=1}^{W_m-2} I_{nmt} \leq V.(1 - z_{W_m mt}) \\ I_{(W_m-1)mt} \geq Q_{mt} + \sum_{n=1}^{W_m-1} I_{nm(t-1)} - D_{mt} - V.(1 - z_{W_m mt}) \\ I_{(W_m-1)mt} \leq Q_{mt} + \sum_{n=1}^{W_m-1} I_{nm(t-1)} - D_{mt} + V.(1 - z_{W_m mt}) \end{cases} \quad \forall m, t \quad (13)$$

$$\sum_{n=1}^{W_m} z_{W_m mt} = 1, \quad \forall\, m, t \quad (14)$$

$$q_{mst} \geq L_{msb} - V(1 - f_{msbt}), \forall\, m, s, b, t \quad (15)$$

$$q_{mst} < U_{msb} + V(1 - f_{msbt}), \forall\, m, s, b, t \quad (16)$$

$$\sum_{b\in B} f_{msbt} = 1, \quad \forall\, m, s, t \quad (17)$$

$$q_{mst} \leq V.F_{mst}, \quad \forall\, m, s, t \quad (18)$$

$$P(q_{mst}) = \sum_{b\in B} f_{msbt}.P_{msb}, \quad \forall\, m, s, t \quad (19)$$

$$\alpha_{mt} \geq \frac{BUD_{mt}}{\sum_{s \in S} P(q_{mst}).F_{mst} + CA_{1mt}}, \quad \forall\, m, t \qquad (20)$$

$$\begin{cases} Q_{mt} \geq -(1 - CA_{1mt}) \\ Q_{mt} \leq V.(1 - CA_{1mt}) \end{cases} \quad \forall m, t \qquad (21)$$

$$\begin{cases} Q_{mt} \geq 1 - V.(1 - CA_{2mt}) \\ Q_{mt} \leq \alpha_{mt} + V.(1 - CA_{2mt}) \end{cases} \quad \forall m, t \qquad (22)$$

$$Q_{mt} > \alpha_{mt} - V.(1 - CA_{3mt}) \quad \forall m, t \qquad (23)$$

$$\sum_{\sigma=1}^{3} CA_{\sigma mt} = 1 \quad \forall\, m, t \qquad (24)$$

$$\gamma_{mt} = (Q_{mt} - \alpha_{mt}).CA_{3mt}, \quad \forall\, m, t \qquad (25)$$

$$CA_{\sigma mt} \in \{0,1\}, \quad \forall\, \sigma, m, t \qquad (26)$$

$$z_{W_m mt} \in \{0,1\}, \quad \forall\, W_m, m, t \qquad (27)$$

$$F_{mst} \in \{0,1\}, \quad \forall\, m, s, t \qquad (28)$$

$$f_{msbt} \in \{0,1\}, \quad \forall\, m, s, b, t \qquad (29)$$

The problem is to find the quantity of each product to be ordered in each period from the suppliers (q_{mst}) while minimizing the total cost, as formulated in Eq. (5). The first term of the objective function indicates total ordering cost, while the second term is used to calculate total purchasing cost. The next term represents the total transportation cost, while the next three terms are used to calculate expiry cost, holding cost, and interest charge.

Eq. (6) is to set no initial and ending inventory. Eq. (7) is to set total purchase quantity of product type *m* in period *t*. In Eq. (8), the number of vehicle from supplier *s* in period *t*, R_{st}, is equal to biggest integer greater than or equal total purchase quantity of all products from supplier *s* in period t, $\sum_{m \epsilon M} q_{mst}$, divide vehicle batch size, *N*.

Eqs. (9)-(14) is linear expressions of product perishability model based on Ref. [11]. Eq. (9) mean the condition when the demand for product type *m* in period *t*, D_{mt}, is less than the inventory of product type *m* with one period of lifetime remaining in that period ($D_{mt} \leq I_{1m(t-1)}$). Eq. (10) mean the condition when the demand for product type *m* in period *t*, D_{mt}, is greater than the inventory of product type *m* with one period of lifetime remaining ($D_{mt} \geq I_{1m(t-1)}$), and less than the sum of the inventories of product type *m* with one and two periods of lifetime remaining ($D_{mt} \leq I_{1m(t-1)} + I_{2m(t-1)}$), and so on. Eq. (14) limit only one of the defined conditions can be occurred in a period time.

Eqs. (15)-(19) are expressions of dynamic inventory problem. Eq. (15) set the lower bound quantity while Eq. (16) set the upper bound quantity in a price break *b* for supplier *s*. Eq. (17) makes sure that a quantity of product type *m* can only be purchased with one single price break *b* from supplier *s* in period *t*. Eq. (18) is to set the purchase quantity of product type *m* from supplier *s* in period *t*, and F_{mst} is a binary variable. Eq. (19) determines the purchase cost of product type *m* per unit, $P(q_{mst})$, based on the quantity purchased from supplier *s* in period *t*.

Eqs. (20)-(25) are expressions of budget availability model. Eq. (20) is to set the maximum number of product type *m* can bought with cash in period *t*. Eqs. (21)-(23) is linear expressions of condition from Eqs. (1)-(3). Eq. (24) is added to ensure that only one of the budget availability constraint sets can be satisfied. Eq. (25) determines the number of product type *m* that bought with credit in period *t*.

5. Solution Procedure

Ref. [13] stated that DLSP is NP-hard problem. Because of that, exact method isn't practical to solve DLSP. In this paper, GA and SA implemented to solve dynamic lot sizing problem with supplier selection considering quantity discount, expiry dates, and budget availability. Fitness function for each solution defined as minimum total cost, as formulated in Eq. (5).

5.1 Genetic Algorithm

Ref. [16] developed GA to solve DLSSP model. Procedure of algorithm in detail is explained as follows:

Step 1. Define parameter.
Define population number (*pop*), crossover probability (ρ_c), mutation probability (ρ_m), and generation number (*k_max*).

Step 2. Define coding scheme.
Every gen in chromosome at Ref. [16] represented purchasing status from supplier *s* in period *t*. Because this paper considered multi-material, chromosome at Ref. [16] developed as shown in Fig 3. Chromosome decoded into three-dimensional matrix (*ḿ* x *ś* x *t́*), where 1 if a purchase of product type m is made from supplier s in period t, and 0 otherwise.

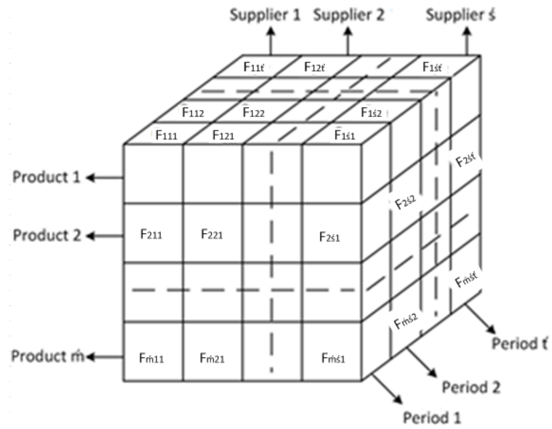

Figure 3. Chromosome of coding scheme

Step 3. Determine initial solution.
For k = 1 generate a set of random number, where:

$$\sum_{s\in S} F_{mst} = 1, \quad \forall\, m, t = 1 \tag{30}$$

$$\sum_{s\in S} F_{mst} \le 1, \quad \forall\, m, t \ge 2 \tag{31}$$

Step 4. Calculate fitness function.
Step 4.1. Determine purchasing status (Y_{mt}).
Define F_{mst} into purchasing status of product type m in period t (Y_{mt}) using Eq. (32).

$$Y_{mt} = \sum_{s\in S} F_{mst}, \quad \forall m, t \tag{32}$$

Step 4.2. Determine total purchase quantity (Q_{mt}).
Define Y_{mt} into purchase quantity of product type m in period t (Q_{mt}) using Eq. (33).

$$Q_{mt} = Y_{mt} D_{mt} + Y_{mt} \begin{bmatrix} (1 - Y_{mt+1}) D_{mt+1} + (1 - Y_{mt+1}) \\ (1 - Y_{mt+2}) D_{mt+2} + \cdots + (1 - Y_{mt+1}) \\ (1 - Y_{mt+2}) \ldots (1 - Y_{mt'}) D_{mt'} \end{bmatrix}, \forall m, t \tag{33}$$

Step 4.3. Determine quantity allocation (q_{mst}).
Define Q_{mt} into quantity allocation for each supplier (q_{mst}) using Eq. (34).

$$Q_{mt} = \sum_{s\in S} q_{mst} F_{mst}, \quad \forall m, t \tag{34}$$

Step 4.4. Calculate total cost
Calculate total cost every chromosome based on quantity allocation (q_{mst}). Best chromosome is chromosome with minimum total cost on population.

Step 5. Crossover operator.
Select a pair of chromosome randomly, then applied two-cut point-crossover to produce two offspring.

Step 6. Mutation operator.
Mutation is used to avoid premature convergence. In this paper, mutation operator is applied by change gen randomly (0 to 1 or 1 to 0).

Step 7. Selection.
After crossover and mutation applied, select chromosome for next generation (k = k + 1). Chromosome was ranked due to minimum fitness function and put into mating pool.

Step 8. Termination.
Repeat *Steps*. 4-8 until $k = k\_max$.

5.2 Simulated Annealing

Ref. [16] developed GA to solve lot sizing problem based on cumulative purchase quantity in Eq. (33). This procedure is irrelevant to developed model. Quantity discount condition make optimal solution can occur when a retailer buy product more than it needed. This paper proposed an algorithm based on Simulated Annealing (SA). Additional notations used in proposed SA are defined as follows:

T_0	initial temperature
T_a	final temperature
T	current temperature
θ_0	initial solution
θ_b	new solution
θ	current solution
θ_a	final solution
$\propto$	cooling rate
k_max	maximum iteration for every temperature increases
K	iteration
$f(\theta_0)$	objective function of initial solution
$f(\theta_b)$	objective function of new solution
$f(\theta)$	objective function of current solution
$f(\theta_a)$	objective function of final solution

Proposed algorithm shown in Fig 5. Procedure of proposed algorithm in detail is explained as follows:
Step 1. Define control parameters.
Define initial temperature (T_0), final temperature (T_a), cooling rate ($\propto$), and maximum iteration (k_max).

Step 2. Determine initial solution.
Initial solution generated randomly using Ref. [16]. Choose solution with minimum total cost.

Step 3. Set current temperature $T = T_0$, set k = 1 and $\theta = \theta_0$.

Step 4. Determine operator modification randomly.
Generate new solution (θ_b) using selected operator modification.

Step 5. Check the feasibility of new solution.
If new solution is feasible go to *Step* 6, otherwise go to *Step* 10. Flowchart of new solution feasibility examination shown in Fig 4.

Step 6. Calculate objective function of new solution $f(\theta_b)$.

Step 7. Calculate $\Delta = f(\theta_b) - f(\theta)$.
If $\Delta <\ 0$ set $\theta = \theta_b$, and go to *Step* 10. Otherwise, go to *Step* 8.

Step 8. Generate random number *r* (0,1).

Step 9. Calculate $e^{(-\Delta/T)}$.
If $r < e^{(-\Delta/T)}$, set solution $\theta = \theta_b$ and go to *Step* 10. Otherwise, go directly to *Step* 10.

Step 10. Check k value.
If $k = k\_max$, go to *Step* 11. Otherwise, set $k = k + 1$ and go to *Step* 4.

Step 11. Set $T = \alpha T$.

Step 12. Set T value.
If $T < T_a$, go to *Step* 13. Otherwise, set $k = 1$ and go to *Step* 4.

Step 13. Set $\theta_a = \theta$, and stop.

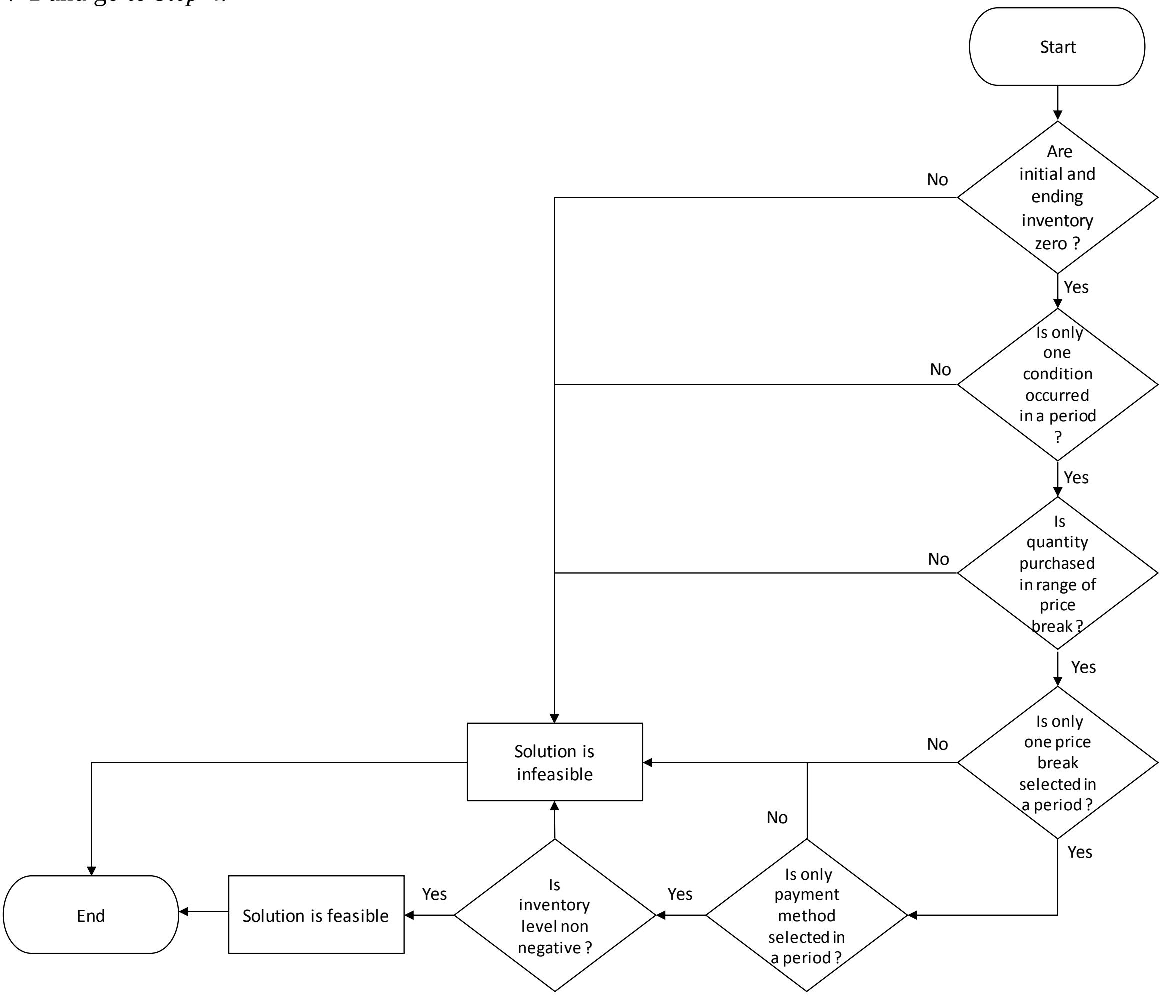

Figure 4. Flowchart of feasibility examination

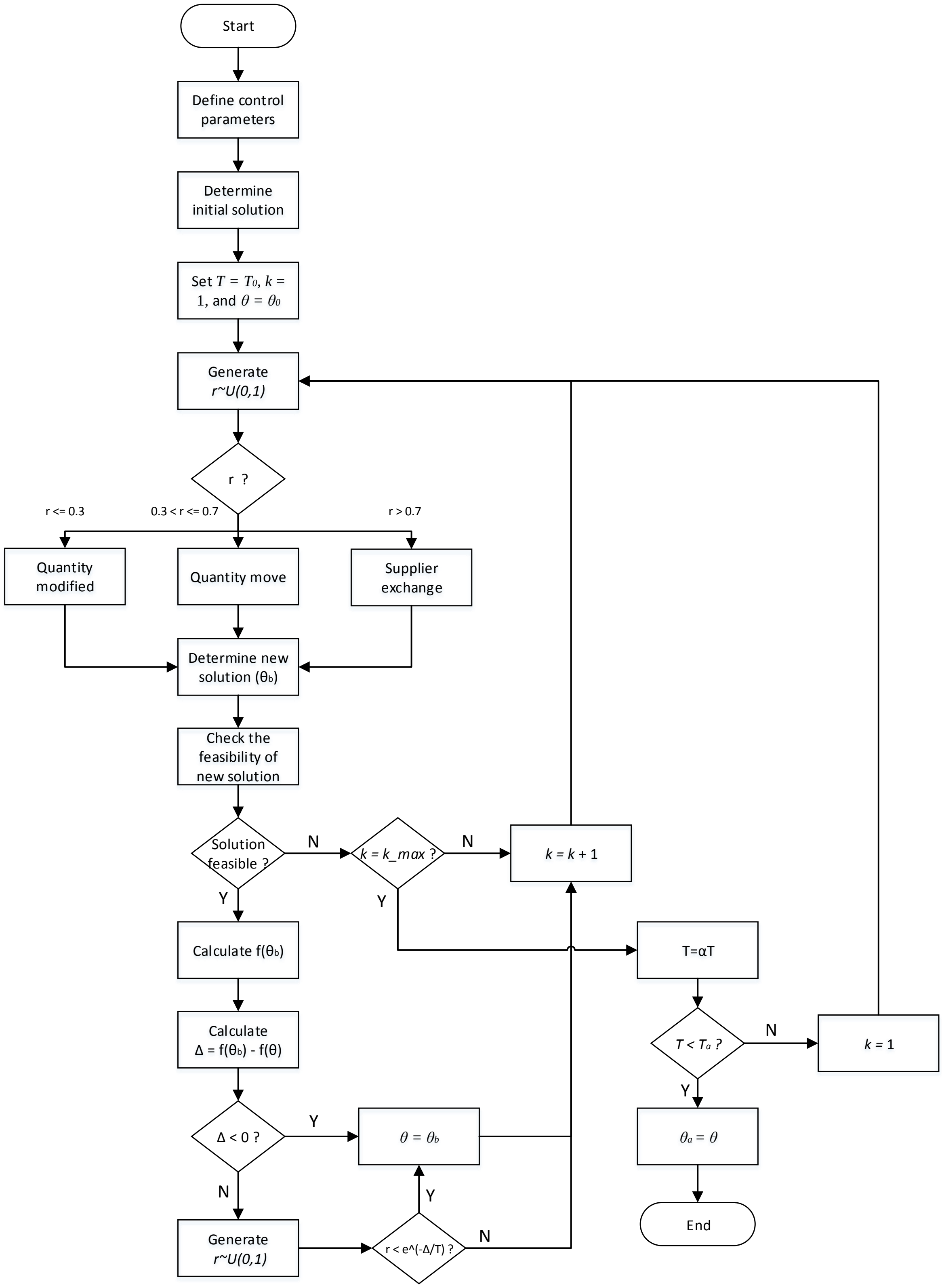

Figure 5. Proposed algorithm

Proposed algorithm used three modification operators, as follows:

1. *Quantity modified*

Quantity modified operator used to modify quantity of product type m purchased in period t (Q_{mt}) randomly. It explores solution with the cheaper purchase cost. Illustration of this operator shown in Fig 6.

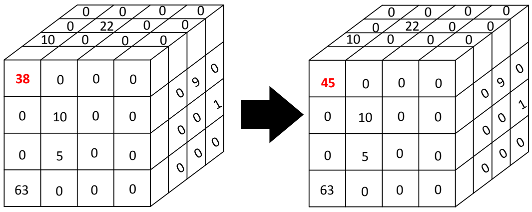

Figure 6. Quantity modified

2. *Quantity moved*

Quantity moved is an operator suggested by Ref. [20]. This operator modifies quantity of product type m purchased in period t (Q_{mt}) and t+1 (Q_{mt+1}). Illustration of this operator shown in Fig 7.

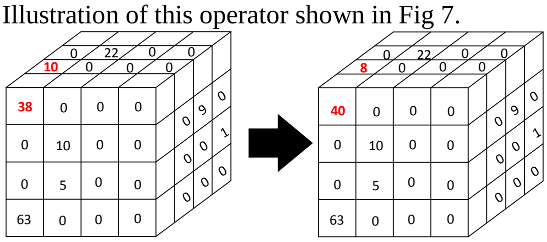

Figure 7. Quantity moved

Procedure of quantity moved in detail is explained as follows:

do {

decide to add or reduce quantity purchased randomly;

if reduce, **then** {

choose product type m in period t randomly;

move quantity as much Δ_1 from period t to period t + 1

}

If add, **then** {

choose product type m in period t randomly;

move quantity as much Δ_2 from period t + 1 to period t

}

where:

$$\Delta_1 = r(0,1) * \sum_{n=1}^{W_m} I_{nmt} \quad (35)$$

$$\Delta_2 = r(0,1) * Q_{mt+1} \quad (36)$$

3. *Supplier exchanged*

Supplier exchanged is an operator to exchange selected supplier with unselected supplier. Illustration of this operator shown in Fig 8.

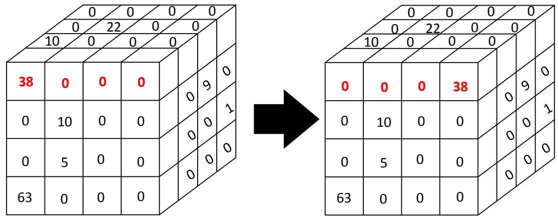

Figure 8. Supplier exchange

Procedure of supplier exchanged in detail is explained as follows:

a. Choose a group of $\sum_{s \in S} F_{mst}$ which has value 1 randomly.
b. Choose one of F_{mst} which has value 0 from that group randomly.
c. Modify chosen F_{mst} to 1, others to 0.

6. Experiments and Discussion

6.1 Parameter Tuning

First step in this paper's experiment is to determine optimal parameter to both metaheuristic algorithms. One of the methods to determine algorithm parameter is using Taguchi method. Taguchi method is fractional experiment method and can be used as alternative to full experiment method [21]. Some response types in Taguchi method is nominal is the best, smaller is better, and larger is better. In this paper, smaller is better used because fitness function is to minimize total cost. Formula for smaller is better:

$$S/N = -10\ x\ Log\left(\frac{S(Y^2)}{n}\right) \quad (37)$$

where:

Y = response

n = orthogonal array number

To demonstrate Taguchi method, first determine level each parameter for both algorithms shown in Table 1. There are 3 levels for each parameter both algorithms. Next, determine fractional experimental design using Minitab 17. Best combination parameter for both algorithms shown in Figs 9-10.

This paper generated numerical example randomly to verified proposed model. Parameter distribution function shown in Table 2. To find near optimal solution, GA and SA coded using MATLAB 17 in PC Intel Core i5 @2.25 GHz RAM 4 GB Operating System Windows 8.1 64-bit. Table 3 show fitness function and CPU time for exact method (LINGO), GA, and SA.

Table 1. Level and parameter both algorithms

Algori-thm	Parameter	Low (1)	Mid (2)	High (3)
GA	Population (A)	50	80	100
	Prob_Crossover (B)	0.6	0.7	0.8
	Prob_Mutation (C)	0.1	0.2	0.3
	Generation (D)	50	80	100
SA	Init_Temp. (A)	10	50	100
	Fin_Temp. (B)	0.01	0.1	1
	Cooling_Rate (C)	0.1	0.5	0.9
	Max_Iteration (D)	50	100	200

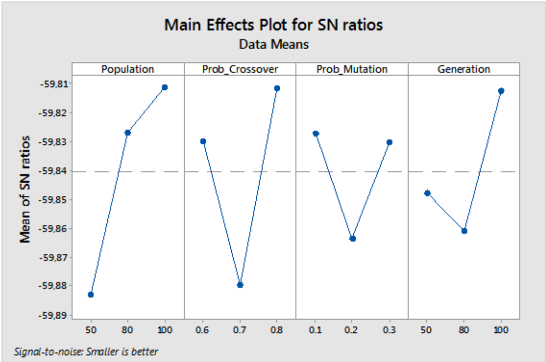

Figure 9. Taguchi S/N ratio plot for GA

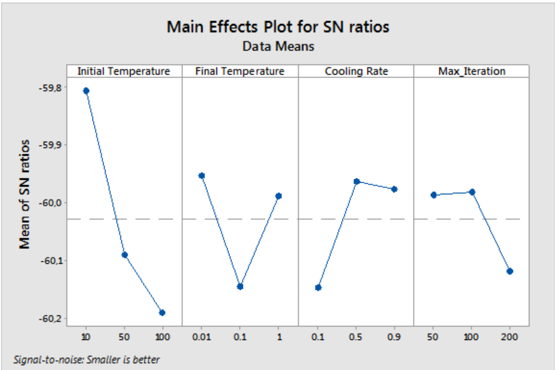

Figure 10. Taguchi S/N ratio plot for proposed SA

Table 2. Parameters of numerical examples

Parame-ters	Value	Parame-ters	Value
D_{mt}	N(50,10)	K_s	Uniform (25,35)
BUD_{mt}	Uniform (200,800)	β_t	Uniform (0.1,0.12)
L_{msb}	Uniform (0,50)	W_m	Uniform (2,4)
P_{msb}	Uniform (3,4)	H_m	Uniform (0.3,0.6)
C_{ms}	Uniform (15,25)	j_m	Uniform (0.5,0.8)

Table 3. Results from proposed model

No	LINGO		GA		SA	
	Z	CPU Time	Z	CPU Time	Z	CPU Time
1	200	0:01:38	230	0:00:23	230	0:00:05
2	201	0:03:22	238	0:00:08	231	0:00:04
3	360	0:04:54	387	0:00:11	368	0:00:04
4	672	0:08:46	702	0:00:25	702	0:00:05
5	201	0:11:08	238	0:00:12	231	0:00:04
6	469	0:27:08	489	0:00:12	472	0:00:05
7	355	0:29:45	383	0:00:20	357	0:00:05
8	468	1:09:42	471	0:00:30	471	0:00:06
9	746	6:32:23	785	0:00:17	785	0:00:04
Av	408	1:58:42	436	0:00:18	428	0:00:05

Running time is reported in the format of h:min:s

6.2 Analysis of Results

Fig 11. show trend of fitness value trend for exact method, GA, and proposed SA. In all problems, proposed SA produce fitness value better than GA. To compare exact method, GA, and proposed SA performances statistically, we use one-way analysis of variance (ANOVA) using MINITAB 17. Table 4 show fitness value LINGO, GA, and proposed SA isn't different statistically.

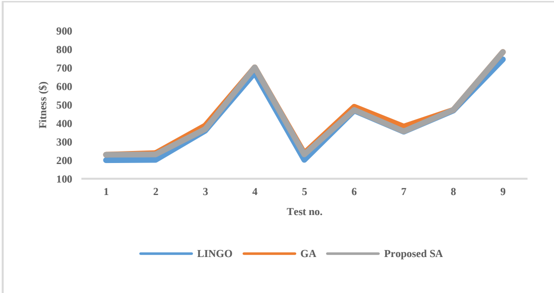

Figure 11. Trend of fitness value

Table 4. Analysis of variance results of fitness value

Source	Methodo-logies	Error	Total
DF	2	24	26
SS	3,656	978,910	982,566
MS	1,828	40,788	
F	0.04		
P value	0.95		

Fig. 12 show trend of CPU time for exact method, GA, and proposed SA. In all problems, both algorithms produce CPU time much smaller than exact method. Table 5 show CPU time of both algorithms different statistically with exact method.

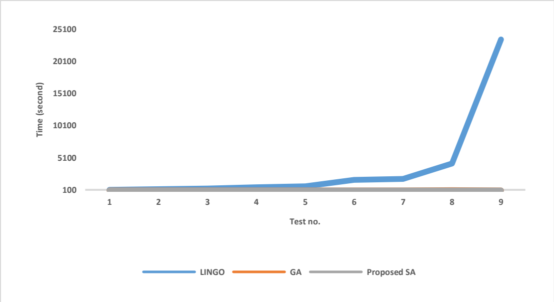

Figure 12. Trend of CPU time

Table 5. Analysis of variance results of CPU time

Source	Methodo-logies	Error	Total
DF	2	24	26
SS	79,810,211	458,002,207	537,812,418
MS	39,905,105	19,083,425	
F	2.09		
P value	0.14		

7. Conclusion

The purpose of this paper is to develop dynamic multi-product multi-period lot sizing model with supplier selection problem with quantity discount, expiry dates, and budget availability to minimize total cost over planning horizon. First, model formulated into mixed integer nonlinear programming, then Genetic Algorithm (GA) and Simulated Annealing (SA) constructed to find near optimal solution. Furthermore, Taguchi method used to calibration of parameter both algorithms. This paper presented numerical example to comparing performance both algorithms. One-way analysis of variance used to compare performance both algorithms statistically. Trend of fitness value show proposed SA has better performance in all problems than GA, but a statistical significant difference was not found. Both algorithms produce CPU time much smaller than exact method.

For future studies, more complex case for supply chain management can be considered. Developed model can considered probabilistic demand, variation in lead time, and stochastic budget availability. Some assumptions can be relaxed. For example, model can consider limited supplier capacity, limited warehouse capacity, and allowed shortage. For addition, model can formulate into multi objective model.

Acknowledgments

The authors would like to thank to Supply Chain Management Department, Institut Teknologi Harapan Bangsa.

References

[1] Chopra, S., Meindl, P., *Supply chain management: strategy, planning, and operation*, Pearson, 2013.

[2] Bowersox, D.J., Closs, D.J., Copper, M.B., Bowersox, J., *Supply chain logistics management*, McGraw-Hill, 2013.

[3] Mendoza, A., Ventura, J.A., "*A serial inventory system with supplier selection and order quantity allocation*", European Journal of Operation Research, Vol. 207, No. 3, pp. 1304-1315, 2010.

[4] Wagner, H.M., Within, T.M., "*Dynamic version of the economic lot size model*", Management Science, Vol. 5, No. 1, pp. 89-96, 1958.

[5] Basnet, C., Leung, J.M.Y., "*Inventory lot-sizing with supplier selection*", Journal of Comp. & Op. Res., Vol. 32, pp. 1-14, 2005.

[6] Pasandideh, S.H.R., Niaki, S.T.A., Mousavi, S.M., "*Two metaheuristics to solve a multi-item multi period inventory control problem under storage constraint and discounts*", Int. J. Adv. Manuf Technol., Vol. 69, pp. 1671-1684, 2013.

[7] Parsa, I., Khiav, M.E., Mazdeh, M.M., Mehrani, S., "*A multi supplier lot sizing strategy using dynamic programming*", Int. J. Industrial. Engineering Comp., Vol. 4, pp. 61-70, 2013.

[8] Mazdeh, M.M., Emadikhiav, M., Parsa, I., "*A heuristic to solve the dynamic lot sizing problem with supplier selection and quantity discounts*", J. Comp. & Ind. Eng., Vol. 85, pp. 33–43, 2015.

[9] Sazvar, Z., Akbari Jokar, M.R., Baboli, A., "*A new order splitting model with stochastic lead times for deterioration items*", J. Syst. Sci., Vol. 0, pp. 1–19, 2013.

[10] Pahl, J., Vob, S., Woodruff, D.L., "*Discrete lot-sizing and scheduling with sequence-dependent setup times and costs including deterioration and perishability constraints*", Proceedings of the 44th Hawaii Int. Conf. on System Sciences, 2011.

[11] Sazvar, Z., Mirzapour Al-e-hashem, S.M.J., Govindan, K., Bahli, B., "*A novel mathematical model for a multi-period, multi-product optimal ordering problem considering expiry dates in a FEFO system*", J. Trans. Res. Part E, Vol. 93, pp. 232–261, 2016.

[12] Katehakis, M.N., Melamed, B., Shi, J., "*Cash-flow base dynamic inventory management*", J.

Prod. & Op. Management., Vol. 0, pp. 1–18, 2016.

[13] Cardenas-Barron, L.E., Gonzalez-Velarde, J.L., Trevino-Garza, G., "*A new approach to solve the multi-product multi-period inventory lot sizing with supplier selection problem*", J. Comp. & Op. Res., Vol. 64, pp. 225–232, 2015.

[14] Ware, N.R., Singh, S.P., Banwet, D.K., "*A mixed-integer non-linear program to model dynamic supplier selection problem*", J. Exp. Sys. With App., Vol. 41, pp. 671–678, 2014.

[15] Zhong, J., Wang, Y., Fan, C., "*Multi-item dynamic lot-sizing model with business volume discount and bounded inventory*", 8th Int. Conf. on BioMedical Engineering and Informatics, 2015.

[16] Lee, A.H,I., Kang, H.Y., Lai, C.M., Hong, W.Y., "*An integrated model for lot Sizing with supplier selection and quantity discounts*", J. Applied Mathematical Modelling, Vol. 37, pp. 4733-4746, 2013.

[17] Nayebi, M.A., Sharifi, M., Shahriari, M.R., Zarabadipour, O., "*Fuzzy-chance constrained multi-objective programming applications for inventory control model*", J. Appl. Math. Sc., Vol. 6, pp. 209-228, 2012.

[18] Mousavi, S.M., Hajipour, V., Niaki, S.T.A., Alikar, N., "*A multi product multi period inventory control problem under inflation and discount : a parameter-tuned particle swarm optimization algorithm*", Int. J. Adv. Manuf. Technol., Vol. 70, pp. 1739-1756, 2014.

[19] Mousavi, S.M., Sadeghi, J., Niaki, S.T.A., Tavana, M., "*A bi-objective inventory optimization model under inflation and discount using tuned Pareto-based algorithms*: NSGA-II, NRGA, and MOPSO", J. App. Soft Computing, Vol. 43, pp. 57-72, 2016.

[20] Ozdamar, L., Barbarosoglu, G., "*An integrated lagrangean relaxation-simulated anneling approach to the multi-level multi-item capacitated lot sizing problem*", Int. J. Production Economics, Vol. 68, pp. 319-331, 2000.

[21] Ross, P.J., "*Taguchi techniques for quality engineering"*, McGraw-Hill, 1988.