Journal of Optimization in Industrial Engineering 7 (2011) 37-44

A Multi-Periodic Multi-Product Inventory Control Problem with Discount: GA Optimization Algorithm

Seyed Mohsen Mousavi[a], Seyed Hamidreza Pasandideh[b,*]

[a] *MSc, Faculty of Industrial and Mechanical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*

[b] *Assistant Professor, Faculty of Industrial and Mechanical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*

Received 5 June, 2010; Revised 23 August, 2010; Accepted 11 September, 2010

Abstract

In this article, a finite horizon, multi product and multi period economic order quantity like seasonal items is considered where demand rate is deterministic and known but variable in each period. The order quantities of items come in batch sizes and the end of the period order quantity and, consequently, demand of customers are zero. In addition, storage space is constrained and the problem was considered under all units discount (AUD) policy. The modeling technique used for this problem is mixed binary integer programming. The objective was to find the minimization optimal order quantities under time value of money over the finite horizon. The inventory control system costs include three costs: ordering cost, holding cost, and purchase cost. In order to solve the proposed model, a genetic algorithm (GA) is applied. Finally, we provide a number of examples in order to illustrate the algorithms further.

Keywords: Inventory control system; All unit discount; Time value of money; Multi-item; Multi-period; Genetic algorithm.

1. Introduction

In real world, the seasonal goods like fashion goods have been attended where they have been ordered at a period (e.g. beginning of season) and have been sold at a given duration (e.g. the season). In multi period inventory control models, the continuous review and the periodic review are the major policies used extensively. However, the basic assumptions of the proposed models restrict their correct usage and utilization in real-world environments. While in continuous review policy the user has the freedom to act at anytime and replenish orders based upon the available inventory level, in the periodic review policy, the user is allowed to reorder the orders only in specific and predetermined times.

Planning for the amount of inventory level and the order time is an important managerial decision that affects the total costs of the inventory system. The most important models ever developed for this decision making is EOQ. Some scholars link the history of EOQ to F.W. Harris [13]. In this research it was necessary to infract some assumptions in order to develop an inventory model for minimizing the total cost of inventory system while there is independent multi-product in the system that the demand for each product is known but inconstant. Furthermore, there is an overall storage constraint and specific all units discount (AUD) for all items or products.

* Corresponding Author Email: Pasandid@yahoo.com

The literature on multi items inventory control problems is scanty. Lee and Kang [18] developed a model for managing an inventory of a product in multiple periods. Their model is fitted for one product. Silver and Mood [28] considered a constrained optimization model for a group of end items with known, constant demand rate and convertibility to other useful units. Das et al. [9] studied a multi-item inventory model with constant demand and infinite replenishment under the restrictions on storage area, total average shortage cost and total average inventory investment cost. Also there are some works on multi-item newsboy problem in literature [3, 1, 6, 2, and 26]. Further, there exist some works that considered multi-period inventory control problems. Kim and Kim [16] formulated a multi-period inventory/distribution planning problem as a mixed integer linear programming and solved it by a Lagrangian relaxation approach. Luciano et al. [19] used value-at-risk (VaR) in the context of inventory management and provided a risk measure for inventory management in a static multi-period framework.

In this article, it is assumed that vendor sells all items under all units discount (AUD) policy. In supply chain, discount quantities can be considered as an inventory coordination mechanism between a buyer and a supplier [27]. Maiti and Maiti [20] developed a problem for multi-item inventory control system of breakable items with

AUD and IQD and a combination of these discounts. Chang and Chang [8] used a linear programming relaxation based on piecewise linearization techniques for solving the inventory problem with variable lead time, crashing cost, and price-quantity discount.

In this research, a mixed binary integer programming technique was used for modeling problem. According to Hillier and Lieberman [14], Binary variables (or 0–1 variables) are used for representing yes-or-no decisions and, consequently, integer programming (IP) problems that contain only binary variables sometimes are called 0-1 integer program or binary integer programming. Ziaee and Sadjadi [29] formulated a model in the form of mixed binary integer programming for solving flow shop scheduling problem under different conditions. Kozanidis and Melachrinoudis [17] developed a branch and bound (B&B) algorithm for the 0-1 mixed integer knapsack problem with linear multiple choice constraints.

The single period problem with non-uniform capacities is known to be weakly NP-hard [11], and [24] proved that multi product inventory control problem is strongly NP-hard. In the present paper, a genetic algorithm (GA) was applied for solving the model. In the recent decades, GA has been used for optimization in different bases. For instance, Michalewicz and janikow [21] used GA for numerical optimization; Guvenier [12] applied a GA approach for inventory classification; and Maiti and Maiti [20] implemented of GA for optimization a discounted multi-item inventory model of breakable items as well as other areas like scheduling (davis [10]), neural network (c.f. Pal et al. [23]), etc.

Most of the traditional inventory models in inventory management literature do not take into account factors like time-value of money. Since the resource of an industry highly correlates with the return of investment and it depends very much on the time of use, therefore, taking the time value of money into account is very critical in managerial decisions. Buzacott [5] was the first who introduced the concept of inflation in inventory modeling. Misra [22] considered internal as well as external inflation rates in his model and analyzed the influence of interest on replenishment strategies. Chandra and Bahner [26], and Sarker and Pan [25] developed infinite/finite replenishment models with shortages, considering inflation and time value of money. Bose et al. [4] developed an economic order quantity model for deteriorating items with linear time dependent demand and shortages, incorporating the effects of inflation and time value of money. In the next section, the assumptions and notation related to this study are presented.

The paper is organized as follows. In section 2, the problem is defined and assumptions and notations are presented. In section 3, the problem is modeled. A genetic algorithm (GA) is applied for solving the model in section 4. Incorporating a numerical example, the solution method is investigated in Section 5. Finally, the conclusion and recommendations for future research come in Section 6.

2. Problem Definition

2.1. Assumptions

The following basic assumptions are made for the proposed models:

1. The demand rate for each item is independent of the others and is constant in a period. However, it can be different in different periods.
2. Each period can place at most one order, which can include or exclude each item.
3. Each item deliver in special batch-size thus, the order quantity of them must be a multiple of a fixed-sized batch. No split-batch is allowed.
4. There are specific discount schedule for each item thus the price of each unit of each item is dependent on the order quantity. All-units discount has been considered.
5. Shortages are not allowed.
6. The initial inventory level of each item is zero.
7. Storage space is limited.
8. Planning horizon is finite and known. In the planning horizon, there are N periods, and the duration of each period is the same.
9. The discount rate regarding the time value of money is known.

2.2. Notations

The following notations are used throughout the paper for i=1, 2, …, m and j=1, 2, … , N:

Tj the total time that elapsed up to and including the jth replenishment cycle for j=0, 1, …, N.

m number of items.

N number of periods.

K number of price break point.

Bi batch size of item i.

Di,j demand of item i at period t.

Hi inventory holding cost of item i, per unit per period.

M a large number.

Oi ordering cost per replenishment (If order places for one or more item in period t, this cost appear in the period).

Qi,j purchase quantity of item i in period j.

$q_{i,k}$ the upper bound quantity of price break point k for item i.

S available storage space.

Si storage space required per unit of item i.

$U_{i,j,k}$ a binary variable, set equal to 1 if item i is purchased with price break k in period j, and 0 otherwise.

Pi,k purchase cost for one unit of item i in with price break k based on the discount schedule with the order quantity Qi,j.

TC total cost of materials in a planning horizon.

Xi,j beginning inventory level of item i in period j. (in j=1, beginning inventory level for all items is zero)
Wi,j a binary variable, set equal to 1 if a purchase of at one unit of item i is made in period j, and 0 otherwise.
r discount rate, representing the time value of money.
M1 a upper bound for Qi,j
Ii(t) inventory level of item i at time t (Tj ≤ t ≤ Tj+1).

3. Model Formulation

Suppose that total transporting space in each period is constrained and equal to M, therefore, track cannot transport more M units for all items in each period. Thus, it can be formulated as follows:

$$\sum_{i=1}^{m} W_{i,j} \le \sum_{i=1}^{m} Q_{i,j} \le M \sum_{i=1}^{m} W_{i,j}$$

$(j = 1, 2, \ldots, N)$ (1)

According to the purpose of modeling, the total cost of materials must minimize in the system which has three segments:
TC = total ordering cost + total holding cost + total purchase cost (2)

Each elements of the equation (2) is calculated in the following:

The binary variable Wi,j determines that whether at least one batch of an item ordered or not. It clear that ordering cost appear when at least one item purchased, therefore:

Total Ordering Cost = $\sum_{i=1}^{m}\sum_{j=1}^{N} O_i W_{i,j}$ (3)

So, the present value of the total ordering cost is

$$\sum_{i=1}^{m}\sum_{j=1}^{N} O_i W_{i,j} e^{-rT_j} \quad (4)$$

In order to calculate holding cost, inventory of item i Ii(t) for (Tj ≤ t ≤ Tj+1) is calculated as follows:

$$I_i(t) = (X_{i,j} + Q_{i,j}) - D_{i,j}(t - T_j) \quad (5)$$

The present value of the holding cost for item i at interval [Tj , Tj+1] is

$$H_i \int_{T_j}^{T_{j+1}} I_i(t) e^{-rt} dt \quad (6)$$

Therefore, the present value of the total holding cost is

$$\sum_{i=1}^{m}\sum_{j=1}^{N-1} H_i e^{-rT_j} \int_{T_j}^{T_{j+1}} I_i(t) e^{-rt} dt \quad (7)$$

In order to solve Eq (6) first must calculate $\int_{T_j}^{T_{j+1}} I_i(t)e^{-rt}dt$ as following:

$$\int_{T_j}^{T_{j+1}} I_i(t)e^{-rt}dt = \int_{T_j}^{T_{j+1}}((X_{i,j}+Q_{i,j})-(D_{i,j}(t-T_j)))e^{-rt}dt = -(\frac{X_{i,j}+Q_{i,j}}{r})(e^{-rT_{j+1}}-e^{-rT_j})+ D_{i,j}(\frac{T_{j+1}}{r}e^{-rT_{j+1}}-\frac{T_j}{r}e^{-rT_j})+D_{i,j}(e^{-rT_{j+1}}-e^{-rT_j})-\frac{D_{i,j}T_{j-1}}{r}(e^{-rT_{j+1}}-e^{-rT_j}) = \frac{e^{-rT_{j+1}}-e^{-rT_j}}{r}D_{i,j}(\frac{1}{r}-T_{j-1}-(\frac{X_{i,j}+Q_{i,j}}{D_{i,j}})+(\frac{T_{j+1}e^{-rT_{j+1}}-T_je^{-rT_j}}{e^{-rT_{j+1}}-e^{-rT_j}})) \quad (8)$$

Therefore, with replacing Eq. (8) in Eq. (7), we get the present value of total holding cost.

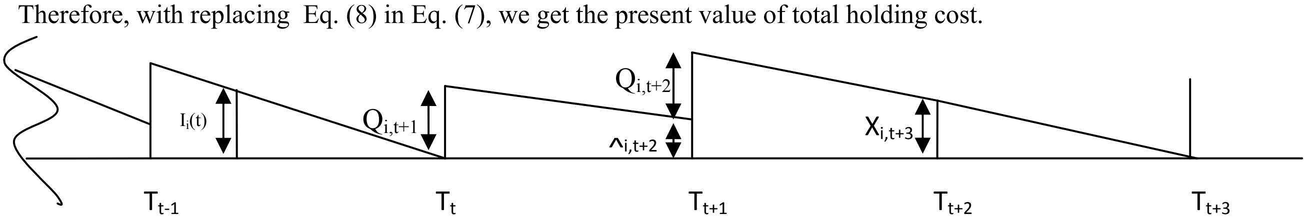

Fig. 1 A graphical representation of all possible cases of the inventory system

The purchase cost is made of cost AUD policy, as explained in the following part:
Therefore, the purchasing cost is offered under the following AUD scheme:

$$P_i = \begin{cases} P_{i,1} & 0 < Q_{i,j} \le q_{i,2} \\ P_{i,2} & q_{i,2} < Q_{i,j} \le q_{i,3} \\ & \vdots \\ P_{i,K} & q_{i,K} < Q_{i,j} \end{cases} \quad (9)$$

The purchasing cost integrated with this policy is obtained as follows:

Total purchasing cost under AUD policy =

$$\sum_{i=1}^{m}\sum_{j=1}^{N-1}\sum_{k=1}^{K} U_{i,j,k} \times \left(P_{i,k} \times Q_{i,j}\right) \quad (10)$$

where that $U_{i,j,k}$ is a binary variable, set equal to 1 if item i is purchased with price break k in period j, and 0 otherwise.

So, the present value of the total purchase cost under AUD policy is

$$\sum_{i=1}^{m}\sum_{j=1}^{N-1}\sum_{k=1}^{K} P_{i,k} Q_{i,j} U_{i,j,k} e^{-rT_j} \tag{11}$$

There are many methods for formulating inventory problems, such as linear programming, nonlinear programming, dynamic programming, geometric programming, gradient-based nonlinear programming, fuzzy geometric programming and mixed integer programming [8]. In this paper, we used mixed binary integer programming for formulating the model.

Assume that $Q_{i,j}$ is the purchase quantity of item i in period j, and $B_i$is the batch size of the item i and$V_{i,j}$ is number of order i-th item in period j. Thus $Q_{i,j}$ calculate as follows:

$$Q_{i,j} = B_i \times V_{i,j} \tag{12}$$

Where, $V_{i,j}$ is an integer variable.

In this model, there are all-units discount tables for each item in all periods. Similar to Lee and Kang [2], the following strategy was selected for exercising this constraint category:

$$q_{i,k} + M \times (U_{i,j,k} - 1) \le Q_{i,j} < q_{i,k+1} + M(1 - U_{i,j,k}) \tag{13}$$

Where $q_{i,k}$ and $q_{i,k+1}$ is the lower and upper bound of quantity in price break k for item i respectively. M is a large number and $U_{i,j,k}$ is a binary variable. If $U_{i,j,k}$ equals zero then $q_{i,k} - M \le Q_{i,j} < q_{i,k+1} + M$ that is true for any $Q_{i,j}$,$q_{i,k}$ and $q_{i,k+1}$. Therefore the constraint will become ineffective. If $U_{i,j,k}$ equals 1, then $q_{i,k} \le Q_{i,j} < q_{i,k+1}$ that means $Q_{i,j}$ is selected from price break point k in discount table and item i.

Given the fact that $Q_{i,j}$ can only be selected from one price break point; therefore, the following constraint must be added: $\sum_{k=1}^{K} U_{i,j,k} = 1$ for (i=1, 2, … , m),(j=1, 2, … ,N). Also, it supposes that the lowest $q_{i,k}$ in AUD table must be zero ($\forall i : q_{i,1} = 0$).

By considering these comments, the model of this problem is as follows:

$$\min TC = \sum_{i=1}^{m}\sum_{j=1}^{N-1} O_i W_{i,j} e^{-rT_j} + \sum_{i=1}^{m}\sum_{j=1}^{N-1} H_i\, e^{-rT_j} \left(\frac{e^{-rT_{j+1}} - e^{-rT_j}}{r} D_{i,j} \left(\frac{1}{r} - T_{j-1} - \left(\frac{X_{i,j} + Q_{i,j}}{D_{i,j}} \right) + \left(\frac{T_{j+1} e^{-rT_{j+1}} - T_j e^{-rT_j}}{e^{-rT_{j+1}} - e^{-rT_j}} \right) \right) \right) + \sum_{i=1}^{m}\sum_{j=1}^{N-1}\sum_{k=1}^{K} P_{i,k} Q_{i,j} U_{i,j,k} e^{-rT_j}$$

Subject to:

$$X_{i,j+1} = X_{i,j} + Q_{i,j} - D_{i,j}$$

$$(i = 1,2, \dots, m), (j = 1,2, \dots, N) \tag{14}$$

$X_{i,1}$=0 (i = 1, 2, … , m)

$$\sum_{i=1}^{m} S_i (X_{i,j} + Q_{i,j}) \le S$$

$$(i = 1,2, \dots, N) \tag{15}$$

$$Q_{i,j} = B_i \times V_{i,j} \quad (i = 1,2, \dots, m), (j = 1,2, \dots, N) \tag{16}$$

$$q_{i,k-1} + M \times (U_{i,j,k} - 1) \le Q_{i,j} < q_{i,k} + M \times (1 - U_{i,j,k})$$

$$(i = 1,2, \dots, m), (j = 1,2, \dots, N), (k = 2,3, \dots, K) \tag{17}$$

$$\textstyle\sum_{k=1}^{K} U_{i,j,k} = 1 \quad (i = 1,2, \dots, m), (j = 1,2, \dots, N) \tag{18}$$

$$\sum_{i=1}^{m} W_{i,j} \le \sum_{i=1}^{m} Q_{i,j} \le M \sum_{i=1}^{m} W_{i,j}$$

$$(i = 1,2, \dots, N) \tag{19}$$

$$Qi\ ,j \le M1 \quad (i = 1,2,\dots,m), (j = 1,2,\dots, N) \tag{20}$$

$$\begin{cases} W_{i,j} \in \{0,1\} & (j = 1, 2, \dots, N) \\ U_{i,j,k} \in \{0,1\} & (i = 1, 2, \dots, m), (j = 1, 2, \dots, N), (k = 1, 2, \dots, K) \end{cases}$$

And all variables are nonnegative.

4. Genetic Algorithm

Many researchers have successfully used metaheuristic methods to solve complicated optimization problems in different fields of scientific and engineering disciplines. The fundamental principle of GAs first was introduced by Holland [30]. Since then, many researchers have applied and expanded this concept in different fields of study.

The proposed model presented here, solved by several metaheuristic approaches, and finally demonstrated that GA is a suitable approach to solve the model. In the next subsection, the basic characteristics of the proposed GA are described.

4.1 Chromosomes

In a GA, a chromosome is a string or trail of genes, which is considered as the coded figure of a possible solution (appropriate or none appropriate). Chromosome representation is a very basic part of the GA method description. In this research, the chromosomes are strings of the order quantity of the products at each period (Qi,j) and are given in real-mode code. For a problem with four products and four periods, the chromosome structure is given in Fig. 2, in which rows are number of the products and columns are number of the periods.

$$Q_2 = \begin{array}{c c} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \end{array}$$

Fig. 2 The presentation of a chromosome

4.2 Population

A batch of chromosomes is called population. Each population or generation of chromosomes has the same size, which is well known as the generation size and is denoted by pop-size. If pop-size is small, then the convergence (algorithm velocity) is faster, but premature convergence may be easily reached by stocking on a local optima. In other words, there will be no diversity in the population, and the population converges too fast. In this research, 40, and 50 are chosen as different population sizes.

4.3 Selection

Selection of chromosomes in mating pool is based on rotate of Roulette Wheel selection method, where chromosomes selection in mating pool is based on their probability selection. So, probability selection of each chromosome is estimated based on its fitness function.

4.4 Crossovers

In a crossover operation, it is necessary to mate pairs of chromosomes to create offspring. Crossover operator procedure is demonstrated below: First with PC probability all chromosomes select for crossover. Then cross two chromosomes at each time in the following way where Q1 and Q2 are parents and Q'1 and Q'2 are offspring's that generated parents. Fig. 3 depicts a uniform crossover operator.

4.5 Mutation

Mutation is the second operation in a GA method for exploring new solutions and it operates on each of the chromosomes resulted from the crossover operation. In mutation, a gene is replaced with other gen randomly where Pm% chromosomes of selection, select for mutation. Fig 4 shows a representation of mutation operator.

4.6 Stopping criterion

Stopping criterion could have different conditions such as achieving to predetermine solution. Average and standard deviation's of solution in algorithm don't any change in several consecutive generations or stop at a certain or predetermined (CPU) time. In this paper, the stopping criteria are achieving to predetermined number of generation.

In short, the steps involved in the GA algorithm used in this research are

1. Setting the parameters Pc, Pm, and pop-size
2. Initializing the population randomly
3. Evaluating the objective function
4. Selecting individual for mating pool by roulette wheel selection method and using elitisms
5. Applying the crossover operation for each pair of chromosomes with probability Pc
6. Applying mutation operation for each chromosome with probability Pm
7. Replacing the current population by the resulting new population
8. Evaluating the objective function
9. If stopping criteria is met, then stop. Otherwise, go to step 5.

$$Q_1 = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ r_{41} & r_{42} & r_{43} & r_{44} \end{bmatrix} \end{array} \qquad Q'_1 = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} a_{11} & a_{12} & r_{13} & r_{14} \\ a_{21} & a_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ r_{41} & r_{42} & r_{43} & r_{44} \end{bmatrix} \end{array}$$

$$\Rightarrow$$

$$Q_2 = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \end{array} \qquad Q'_2 = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} r_{11} & r_{12} & a_{13} & a_{14} \\ r_{21} & r_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \end{array}$$

Fig. 3. The uniform crossover operator

$$Q = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \end{array} \Rightarrow Q' = \begin{array}{cc} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} b_{11} & b_{12} & a_{13} & a_{14} \\ b_{21} & b_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \end{array}$$

Fig. 4 The uniform mutation operator

In order to demonstrate the proposed GA algorithm and evaluate its performances, in the next section.

5. Numerical Examples

In this paper, in order to display the accuracy of the model, 10 numerical examples are considered where the best amounts of parameters in GA are obtained a lot of running for different parameters. Table 1 presents values of examples and properties for the 10 examples. For all examples, a proper statistical design is applied for tuning GA parameters (PC, Pm and Pop-size). Moreover, in order to reduce run time and reduce selection chance of chromosomes that do not satisfy constraints, a penalty function is applied. For constructing penalty function, we suppose there is a constraint as the following:

$$AG(x) \leq B \tag{21}$$

If ∀ x equation (13) is confirmed, the value of its penalty is zero. Otherwise, the value of penalty is calculated as follows:

$$penalty = (AG(x) - B)^4 \tag{22}$$

Table 1
The general data

problem no.	m	N	PC	Pm	pop-size	GEN
1	4	4	0.68	0.2	40	500
2	4	5	0.68	0.2	40	500
3	4	7	0.64	0.2	40	1000
4	3	4	0.65	0.25	40	500
5	6	6	0.64	0.2	50	1000
6	4	6	0.64	0.15	40	500
7	6	4	0.68	0.15	40	500
8	6	5	0.64	0.2	40	500
9	6	7	0.65	0.25	50	1000
10	6	8	0.65	0.2	50	1000

Table 2
The data for problem of 4 items and 4 periods

i	1	2	3	4
Di,1	30	100	80	30
Di,2	40	70	25	90
Di,3	73	83	102	40
Di,4	0	0	0	0
Bi	3	7	5	8
Hi	3	2	1	4
Oi	5	15	12	11
Si	4	6	7	5

Similar to (22), a penalty function is defined for the problem constraints and added to objective function. Table 2 shows the data for an example with 4 products and 4 period and tables 3 and 4 display the results of GA for different examples in term of objective values (for all example) and order quantities.

Table 3
The order quantities for problem with 4 items and 4 periods

i	1	2	3	4
Qi,1	33	105	105	32
Qi,2	39	70	0	88
Qi,3	72	84	105	40

(The problem coded with MATLAB 7.8.0.347 (R2009a) software. GA is coded in real code representation using Roulette wheel selection, uniform crossover and single point mutation operators randomly.

Fig. 5 displays the convergence path of the best results by Genetic algorithm for the problem with 4 items and 4 periods.

6. Conclusion

In this paper, we considered a mixed binary integer programming for a problem of multi items and multi periods and all units discount so that the space of warehouse is constrained.In this research, shortages are not allowed. It is assumed that the order quantities constrained with an upper bound and order quantities are zero at the end of the period and the demands of items at the end of period are zero because the planning horizon is finite and known (like seasonal items). The presented model in this paper is different from the traditional inventory models in that it embraces a binary variable that is applied for order quantity for each item in each period as if we generate an order for an item in a period; it is 1, otherwise, it is zero. In order to solve the problem, we applied a genetic algorithm using a roulette wheel method for selection, a uniform method for crossover, a single point mutation randomly, and a predetermined number of generations for stopping criterion. 5 numerical examples are generated to demonstrate accuracy of GA.

Table 4
The best results of objective function for GA

problem no.	1	2	3	4	5	6	7	8	9	10
objective function	8823	10999	14137	6765	37125	12530	21472	314226	64126	80549

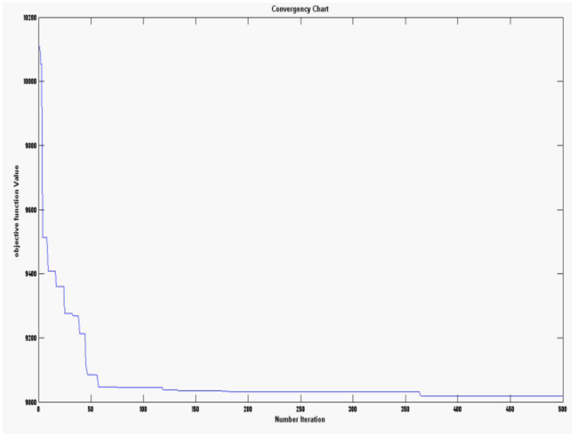

Fig. 5. The convergence path of the best result by Genetic algorithm

7. References

[1] L. L. Abdel-Malek, R. Montanari, An analysis of the multi-product newsboy problem with a budget constraint. International Journal of Production Economics. 97(3): p. 296-307. 2005.

[2] L. L. Abdel-Malek, R. Montanari, on the multi-product newsboy problem with two constraints. Computers & Operations Research. 32(8): p. 2095-2116. 2005.

[3] L. Abdel-Malek, R. Montanari, and L.C. Morales, Exact, approximate, and generic iterative models for the multi-product Newsboy problem with budget constraint. International Journal of Production Economics. 91(2): p. 189-198. 2004.

[4] S. Bose, A. Goswami, K. S. Chaudhuri, An EOQ model for deteriorating items with linear time-dependent demand rate and shortages under inflation and time discounting, Journal of Operational Research Society, 46. 771-782. 1995.

[5] J. A. Buzacott, Economic order quantities with inflation, Operational Research Quarterly, 26. 553-558. 1975.

[6] R. J. Casimir, The value of information in the multi-item newsboy problem. Omega. 30(1): p. 45-50. 2002.

[7] M. J. Chandra, M. L. Bahner, The effects of inflation and the time value of money on some inventory systems, International Journal of Production Research, 23. 723-730. 1985.

[8] C. T. Chang, S.C. Chang, on the inventory model with variable lead time and price-quantity discount. Journal of the Operational Research Society. 52(10): p. 1151-1158. 2001.

[9] K. Das, T. K. Roy, and M. Maiti, Multi-item inventory model with quantity-dependent inventory costs and demand-dependent unit cost under imprecise objective and restrictions: a geometric programming approach. Production Planning & Control. 11(8): p. 781-788. 2000.

[10] L. Davis (Ed.). Hand book of Genetic Algorithms, NewYork, Van Nostrand Reinhold, 1991.

[11] M. Florian, J. K. Lenstra, and A. H. G. Rinooy Kan. Deterministic production planning: Algorithms and complexity. Management Science, 26:669-679, 1980.

[12] H. A., Guvenir, A Genetic Algorithm for Multicriteria Inventory Classification in Artificial Neural Nets and Genetic Algorithms, Proceedings of the International Conference, Ales, France, D. W. Pearson, N. C. Steele, and R. F. Albrecht (Eds), Springer-Verlag, Wien.6-9. 1995.

[13] F. W. Harris, How many parts to make at once? Factory, the Magazine of Management, 10- 135-152. 1913.

[14] Hillier/Lieberman, Introduction to Operations Research, McGraw-Hill.2001.

[15] J. H. Holland. Adaptive in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, 1975.

[16] J. U. Kim, Y. D. Kim, a Lagrangian relaxation approach to multi-period inventory/distribution planning. Journal of the Operational Research Society. 51(3). 2000.

[17] G. Kozanidis, E. Melachrinoudis, A branch & bound algorithm for the 0-1 mixed integer knapsack problem with linear multiple choice constraints. Computers & Operations Research. 31(5): p. 695-711. 2004.

[18] A. H. I., Lee, and H.-Y. Kang, A mixed 0-1 integer programming for inventory model a case study of TFT-LCD manufacturing company in Taiwan. Kybernetes. 37(1): p. 66-82. 2008.

[19] E. L., Luciano, Peccati, and D.M. Cifarelli, VaR as a risk measure for multiperiod static inventory models. International Journal of Production Economics. 81-2: p. 375-384. 2002.

[20] A. K. Maiti, M. Maiti, Discounted multi-item inventory model via genetic algorithm with Roulette wheel selection, arithmetic crossover and uniform mutation in constraints bounded domains. International Journal of Computer Mathematics. 85(9):p.1341-1353. 2008.

[21] Z. Michalewicz, Janikow, C. Z.: Genetic Algorithms for Numerical Optimization. Statistics and Computing, 1. 75–91. 1991.

[22] R. B. Misra, A study of inflationary effects on inventory systems, Logist. Spectrum, 9. 260-268. 1975.
[23] S. K. Pal, S. De, and A. Ghosh. Designing Hopfield type networks using genetic algorithms and its caparisons with simulated annealing, Int. J. Patterns Recognit. Artif. Intell. 11, pp. 447–461. 1997.
[24] A. Retsef Levi, Lodi and M. Sviridenko. Approximation Algorithms for the Capacitated Multi-Item Lot-Sizing Problem via Flow-Cover Inequalities
[25] B. R. Sarkar, Pan, H., Effects of inflation and time value of money on order quantity and allowable shortage, International Journal of Production Economics, 34. 151-155. 1994.
[26] Z., Shao, X. Ji, Fuzzy multi-product constraint newsboy problem. Applied Mathematics and Computation. 180(1): p. 7-15. 2006.
[27] H. J., Shin, W.C. Benton, Quantity discount-based inventory coordination: Effectiveness and critical environmental factors. Production and Operations Management. 13(1): p. 63-76. 2004.
[28] E. A., Silver, I. Mood, Multi-Item Economic Order Quantity Model with an Initial Stock of Convertible Units. Engineering Economist. 46(2): p. 129. 2001.
[29] M., Ziaee, S.J. Sadjadi, Mixed binary integer programming formulations for the flow shop scheduling problems. A case study: ISD projects scheduling. Applied Mathematics and Computation. 185(1): p. 218-228. 2007.