International Journal of Mathematical Archive-8(5), 2017, 13-22
Available online through www.ijma.info ISSN 2229 – 5046

FUZZY OPTIMAL PRODUCTION INVENTORY CONTROL

BAYOUMI. M.ALI HASSAN*, EL-ALAOUI T. M**.

*Faculty of Computer and Information, Cairo University.

**Dept. of Mathematics, Faculty of Science, Girl's Branch, El-Azhar University.

(Received On: 24-06-15; Revised & Accepted On: 19-05-17)

ABSTRACT

In this paper, a fuzzy optimal multi-level production–inventory control model is formulated as optimal control problem in a decentralized supply chain having individual decision makers (DMs) where each facility is controlled independently in a continuous review. The adopted approach is the Triangular Fuzzy Numbers (TFNs) utilization. Accordingly, the genetic algorithm is developed, coded with C^{++} Programming language. Numerical example and sensitivity analysis are discussed to illustrate the optimal decision in fuzzy environments. This paper introduce a method so that the production rate that the manufacturer should hold along the chain such that the resources of each company are used the best and the inventory overall cost is to be minimized?

Key words: *Fuzzy Optimal control; supply chain management (SCM); Chebyshev Approximation; Genetic Algorithm.*

1. INTRODUCTION

In most of the existing inventory models, it is assumed that the inventory parameters, objective goals are deterministic and fixed. But, if we think of their practical meaning, they are uncertain. The fuzzy criterion models are just able to incorporate the expert knowledge via fuzzy membership functions. So, fuzzy criterion models are closer to the spirit of modern decision-making, thinking (see Turban 1998) , than the existing inventory models. In view of the fuzziness of goals, constraints and actions in real-world, decision making problems such as inventory control systems and service facility systems, metaheuristic algorithms (see Thangavel 2005) and fuzzy dynamic programming have been widely applied. In 1870, Bellman and Zadeh considered the classical decision model and suggested several models for decision making in a fuzzy environment. Bellman .R.E. and L.A.Zadeh (see Bellman 1970 and Zimmermann 1985) deal with the application of fuzzy set theory in mathematical programming. (See Kacprzyk 1983, Esogbue and Bellman 1984 and Zimmerman 1983, 1991 and 2000) reviewed development and applications in the field of fuzzy inventory control.

In this paper, the imprecision and uncertainty are described in three cases:

A. When production costs are presented as Triangular Fuzzy Numbers (TFNs).
B. When holding costs are presented as Triangular Fuzzy Numbers (TFNs).
C. When set-up costs are presented as Triangular Fuzzy Numbers (TFNs).
Throughout the supply chain described by linguistic terms.

2. MODEL DESCRIPTION

Assumptions and Notations

For the Nth (N =1, 2,…, n_1) item, it is assumed that:

(i) Initial stock and demands are known.
(ii) Demand is continuous and deterministic.
(iii) Production is continuous and deterministic.
(iv) Inventory level is continuous and deterministic.
(v) Deterioration rate is known and deterministic.
(vi) Backlog or shortage is not considered.

Corresponding Author: Bayoumi. M.Ali Hassan,*
**Faculty of Computer and Information, Cairo University. E-mail: dr_bayoumi10@fci-cu.edu.eg*

For the Nth (N=1, 2,…,n$_1$) item :

$D_N(t) = D_0(t)+D_{1N}(t)$: demand rate at time t where D_{0N} , D_{1PN} are known.

Pn(t) = $\sum_{K=0}^{m} P_{kN}.T_k(t)$ **:** Production rate at time t,

where P_{0N}, P_{1N},…,P_{mN} are known and $T_K(t)$ is the Kth Chebyshev polynomial .

$I_{iN}(t)$ **:**inventory level at time t in the ith cycle.

$\tilde{C}_{pN}$ **:** Fuzzy production cost per unit item and time represented as TFN.

$\tilde{h}_N$ **:** Fuzzy holding cost per unit item and time represented as TFN.

a$_N$: Storage area for the Nth item.

$\tilde{S}_p$ **:** Fuzzy set up time per echelon represented as TFN.

[t$_{i-1}$,t$_i$] **:** the interval of the ith cycle, where i= 1,2,…, ℓ.

ζ$_s$ **:** is the Chebyshev points in the interval [-1,1] **,** where s = 0, 1, 2,…,j.

where:

ℓ Level number.
T Time length of each level.
n$_1$ number of items.
M maximum space available for storage
B$_T$ Total budget.

Defuzzification of parameters

The triangular fuzzy number TFN is computed using the equation (A).

$$\hat{A}=\frac{a_1+2a_2+a_3}{4} \tag{A}$$

where

A = (a$_1$, a$_2$, a$_3$) is a TFN.

Parameter 1: $\tilde{C}_{pN}$ is a Fuzzy production cost per unit item and unit time represented as TFN

Parameter 2: $\tilde{h}_N$ is a fuzzy holding cost per unit item and unit time represented as TFN

Parameter 3: $\tilde{S}_p$ is a fuzzy set up time per cycle represented as TFN.

Fuzzy parameters	TFN	Defuzzification
$\tilde{C}_{pN}$	(c$_1$, c$_2$, c$_3$)	$C_{pN}=\frac{c_1+2c_2+c_3}{4}$
$\tilde{h}_N$	(h$_1$, h$_2$, h$_3$)	$h_N=\frac{h_1+2h_2+h_3}{4}$
$\tilde{S}_p$	(s$_1$, s$_2$, s$_3$)	$S_p=\frac{s_1+2s_2+s_3}{4}$

Table-1: Defuzzification of Parameters

Mathematical formulation:

This is a serial decentralized supply chain system with individual decision makers (DMs) where each facility is controlled independently in a continuous review.

The overall SC inventory control is achieved at two levels consisting of Producers or manufacturers, a central warehouse and distributors or retailers. (Figure 1)

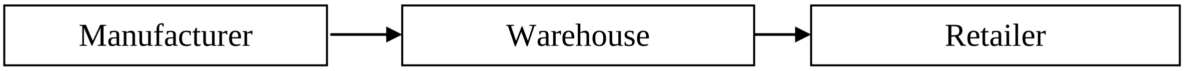

Figure-1: Serial supply chain system

Each one of them holds inventory in some form to support the requirement at the end of the supply chain. This problem is formulated as an Optimal Control Problem.

We assume that the system is multi-item production with dynamic demands. Here, the items are produced at a variable rate and deteriorate at a constant rate. Demands for the items are time dependent and the stock level at time t decreases due to deterioration and consumption. Shortages are not allowed.

The imprecision and uncertainty are described in three cases:

a. When production costs are presented as Triangular Fuzzy Numbers (TFNs).
b. When holding costs are presented as Triangular Fuzzy Numbers (TFNs).
c. When set-up costs are presented as Triangular Fuzzy Numbers (TFNs).

The differential equation for items representing the above system during a fixed time-horizon, T, are given by:

$$\text{Min } J = \int_{N=1}^{n1} \sum_{i=1}^{n} [h_N I_N(t) + C_{pN} P_N(t)]dt + ns_p \tag{1.a}$$

$$= \sum_{N=1}^{n1}\sum_{i=1}^{n}\int_{t_{i-1}}^{t_{iN1}}[h_N I_{iN}(t) + C_{pN} P_N(t)]dt + \sum_{p=1}^{n1}\sum_{i=1}^{n}\int_{t_{ip1}}^{t_i} h_N P_{iN}(t)\,dt + ns_p \tag{1.b}$$

Subject to

$$\frac{dI_{iN}(t)}{dt} = \mathrm{P_N(t) - D_N(t) - \theta_N\, I_{iN}\,(t)},\ t_{i-1} \le t \le t_{iN1} \tag{2.a}$$

$$\frac{dI_{iN}(t)}{dt} = -\,\mathrm{D_N(t) - \theta_N\, I_{iN}\,(t)},\ t_{iN1} \le t \le t_i \tag{2.b}$$

$$\sum_{N=1}^{n1} I_{iN}(t)a_N \le \mathrm{M} \tag{3}$$

$$\sum_{i=1}^{n}\sum_{p=1}^{n1}\int_{t_{i-1}}^{t_{iN1}} C_{pN} P_N(t)dt + ns_p \le \mathrm{B}_T \tag{4}$$

and

$$\mathrm{J_1} = \sum_{N=1}^{n1}\sum_{i=1}^{n}\int_{t_{i-1}}^{t1}[h_N I_{iN}(t) + C_{pN} P_N(t)]dt \text{ and } \mathrm{J_2} = \sum_{p=1}^{n1}\sum_{i=1}^{n}\int_{t_{ip1}}^{t_i} h_N P_{iN}(t)dt \tag{5}$$

With the initial and boundary conditions:

$\mathrm{D_N\,(t) = D_{0N}+D_{1N}\,t}$
$\mathrm{I_{iN}(t_i) = I_{iN}\,(t_{i-1}) = 0}$ and
$t_i = iT/n$ where $i = 1, 2,\ldots,n_1$ (6)

The problem can be converted from the t- interval $[t_{i-1,}\ t_{ip1}]$ into the ζ –interval [0, 1], via the transformation

$$\zeta = \frac{t - t_{i-1}}{t_{iN1} - t_{i-1}} \tag{B}$$

So, equation (8) is reduced to:

$$\mathrm{J_1} = \sum_{N-1}^{n_1}\sum_{i-1}^{n}(t_{iN1} - t_{i-1})\int_0^1 [h_N I_{iN}(\xi) + C_{pN} P_N(\xi)]d\xi \tag{7}$$

and

$$\frac{dI_{iN}(\xi)}{d\xi} = (t_{ip1}\text{-}\ t_{i-1})\ (u_p\,(\xi) - D_{0N} - D_{1N}\,((t_{\ iN1}\text{-}t_{\ i-1}\,)\ \xi + t_{i-1}\,) - \theta_N\, I_{\ iN}\,(\xi) \tag{8}$$

Where

$$\mathrm{I_{iN}\,(0) = 0} \qquad \text{and} \qquad P_N(\xi) = \sum_{k=0}^{m} P_{kN} T_k(\xi) \tag{9}$$

Assume

$$\frac{dI_{iN}(\xi)}{d\xi} = \Phi_{iN}(\xi) \qquad (10)$$

$$I_{iN}(\xi_s) = \sum_{j=0}^{N} b_{sj}\Phi_{iN}(\xi_j) \qquad (11)$$

Where b_{sj} are the elements of the matrix as given in (El gendi 1969) method:

$$\sum_{j=o}^{k} b_{sj} = \left(1-\cos\frac{s\pi}{k}\right), \qquad s = 0, 1,\ldots,k \text{ and } i = 1, 2\ldots,n$$

Here: $P_{00}, P_{01},\ldots,P_{mn1}$ are unknown constants and $T_k(\xi)$, $k = 0, 1, 2\ldots,m$ is the kth Chebyshev polynomial .

Using equations (9, 10, 11), the system of constraints (8), can be written in the following form

$$\Phi_{iN}(\xi_s) = (t_{iN1}-t_{i-1})\,(\sum_{k=0}^{m} P_{kN}T_k(\xi) - D_{0N} - D_{1N}\ ((t_{iN1}-t_{i-1})\,\xi_s + t_{i-1}) - \theta_N \sum_{j=0}^{N} b_{sj}\Phi_{iN}(\xi_j) \qquad (12)$$

and $$\xi_s = ½\left(1-\cos\frac{s\pi}{k}\right) \quad s = 0,1,\ldots,k \qquad (13)$$

Equation (7) can be approximated using El-Hawary technique (see El-Gindy and El-Hawary 1995) and substituting from (9,10,11, 12) into (7) we get:

$$J_1(P_{00},P_{01},\ldots,P_{mn1}) = \sum_{N=1}^{n1}\sum_{i=1}^{n}\left(t_{iN1}-t_{i-1}\right)\left[\sum_{j=0}^{k} b_{kj}(h_N\sum_{s=0}^{k} b_{js}\Phi_{iN}(\xi_s) + C_{pN}(\sum_{k=0}^{m}P_{kN}T_k(\xi_j)))\right] \qquad (14)$$

Integrating (2.b) and the 2nd part of (5) using (7) we get

$$I_{iN}(t) = D_{0N}(\exp(\theta_N(t_i-t))-1)/\theta_N + D_{1N}(ti\exp(\theta_N(t_i-t))-t)/\theta_N$$
$$- D_{1N}(\exp(\theta_N(ti-t))-1)/\theta^2_N \qquad t_{iN1} \le t \le t_i \qquad (15)$$

$$J_2 = \sum_{N=1}^{n1} h_N \sum_{i=1}^{n}[D_{0N}(\exp(\theta_N(t_i-t_{iN1}))-1)/\theta_N{}^2 - D_{0N}(t_i-t_{iN1})/\theta_N$$
$$- D_{1N}(t_i{}^2 - t_{iN1}{}^2)/2\theta_N{}^2 - (D_{1N}\exp\theta_N(t_i-t_{iN1})-1)/\theta_N{}^3$$
$$+ D_{1N}(t_i-t_{iN1})/\theta_N{}^2 + D_{1N}(t_i\exp\theta_N(t_i-t_{iN1})-t_i)/\theta_N{}^2] \qquad (16)$$

To find the values of t_{iN1}, equating equations (13) and (17), we get:

$$D_{0N}(\exp(\theta_N(t_i-t))-1)/\theta_N + D_{1N}(ti\exp(\theta_N(t_i-t))-t)/\theta_N$$
$$- D_{1N}(\exp(\theta_N(ti-t))-1)/\theta^2_N = \sum_{j=0}^{N} b_{kj}\Phi_{iN}(\xi_j),\ i=1, 2,\ldots,n,\ N= 1, 2,\ldots,n_1 \qquad (17)$$

Then the above optimization problem can be written in the following form:

Min $J = J_1 + J_2 + n\,S_p$ (18)

Subject to

$$\sum_{N=1}^{n1}\ [\,(t_{iN1}-t_{i-1})\,(\sum_{j=0}^{k} b_{kj}\sum_{k=0}^{m}P_{kN}T_k(\xi_j))\,] \le M \qquad (19)$$

$$\left[\sum_{N=1}^{n1}\sum_{i=1}^{n}\left(t_{iN1}-t_{i-1}\right)\left(\sum_{j=0}^{k} b_{kj}\sum_{k=0}^{m}P_{kN}T_k(\xi_j)\right)C_{pN} + ns_p\right] \le B_T \qquad (20)$$

This is a nonlinear programming problem (NLPP) of the performance with a system of constraints, and we solve it using the Genetic algorithm (GA).

This is a fuzzy Non-Linear optimization problem of the performance index subject to some constraints.

We can now defuzzify the parameters given by a linguistic expression and represented as triangular fuzzy numbers (TFN_s) as follows:

a. Fuzzy production cost $\left(\tilde{C}_{pN}\right)$:

The values of the production cost parameter C for our algorithm can be determined from the production engineer experiences to be as follows:

C_1 =8, C_2= 10 and C_3=15

$\tilde{C}_{pN}$= (8, 10, 15) is a TFN.

So, $C_{pN} = \frac{c_1 + 2c_2 + c_3}{4}$ **(A)**

To defuzzify our model we must substitute the fuzzy value of the production cost by its crisp value using equation (A)

b. Fuzzy holding cost $\left(\tilde{h}_N\right)$:

The values of the cost parameter C for our algorithm can be determined from the finance manger experiences or the person which has the responsibility of inventory to be as follows:

C_1 = 7, C_2 = $_8$ and C_3= 12 and so $\tilde{C}_{pN}$= (7, 8, 12) is a TFN

So, $h_N = \frac{c_1 + 2c_2 + c_3}{4}$ **(A)**

To defuzzify our model we must substitute the fuzzy value of the holding cost by its crisp value using equation (A), so h_N = 8.75 units.

c. Fuzzy setup cost $\left(\tilde{S}_p\right)$

The values of the set up cost C for our algorithm can be determined from the finance manger experiences to be as follows:

C_1 =5, C_2= 6 and C_3= 9

$\tilde{S}_{pN}$= (5, 6, 9) is a TFN

$S_{p_N} = \frac{c_1 + 2c_2 + c_3}{4}$ **(A)**

To defuzzify our model we must substitute the fuzzy value of the setup cost by its crisp value using equation (A), so S_{p_N} = 6.5 units

Now the reduced Non-Linear constrained optimization problem can be solved

Using a Genetic algorithm (GA).

3. GENETIC ALGORITHM

A genetic algorithm (GA) is a heuristic search process for optimization that resembles natural selection. The GA was first proposed by Holland. It has been applied successfully in different areas.

Genetic Algorithms are a class of optimization algorithms based on “survival of the fittest”. The basic idea is that each possible solution is a member of a population, and any given population is keeping track of multiple solutions. When going through a genetic algorithm a good solution is more likely to survive and hence more likely to reproduce.

Parents in a genetic algorithm are selected at random from the available population, and the new trial solutions (children) are created from the parents. When these children are added to the population, they occasionally have mutations which add more variety to the population.

The ideas behind the genetic programming paradigm stem from the original work of John Holland on genetic algorithms. It has been documented that by using genetic algorithms, one can search a solution space in non-linear time, with relative certainty that the outcome will converge on a near-optimal, global solution.

Genetic programming operators

The four general genetic operators are: crossover, reproduction, mutation, and inversion. The following overview describes each in brief.

Crossover, considered along with reproduction, to be the two foremost genetic operations. It is mainly responsible for the genetic diversity in the population of programs. Similar to its performance under genetic algorithms, crossover operates on two programs (a binary operator), and produces two child programs. Two random nodes are selected from within each program and then the resultant ``sub-trees" are swapped, generating two new programs. These new programs become part of the next generation of programs to be evaluated.

Reproduction, the second main operation, is performed by simply copying a selected member from the current generation to the next generation.

In genetic programming, when an individual incestuously mates with itself (or copies of itself), the two resulting offspring will, in general, be different as before, the Darwinian reproduction operation creates a tendency toward convergence; however, in genetic programming, the crossover operation exerts a counterbalancing pressure away from convergence. Thus, convergence of the population is unlikely in genetic programming.

Mutation With genetic algorithms, mutation becomes an important operator which provides diversity to the population. However, mutation is relatively unimportant in the new environment, because the dynamic sizes and shapes of the individuals in the population already provide diversity, and as stated above, the population should not converge. Thus, mutation can be considered as a variation on the crossover operation.

Inversion (permutation) the effectiveness of this operation has never been conclusively demonstrated.

GA Implementation

It is generally accepted that a GA to solve a decision making problem must have five basic components:

1. Values for the parameters (population size, probabilities of applying genetic operators, etc.),
2. Genetic representation for potential solutions,
3. A way to create an initial population of solutions,
4. An evaluation function (i.e., the environment), rating solutions in terms of their “fitness”, and
5. Genetic operators that alter the genetic composition of parents during reproduction.

Assigning a fitness value

We must determine how good the individuals are at solving the given problem. And as with genetic algorithms, the crossover and reproduction operations are separate from the actual evaluation of the fitness, making the genetic programming operators problem-independent.

The measurement of fitness is a rather nebulous subject. Since, it is highly problem-dependent, we consider massaging the results to make fitness evaluation much easier, through a process known as *scaling*. Simply put, scaling standardizes the measurement of how fit a particular individual is with respect to the rest of the population. Based on the fitness value, we go about this selection for survival in one of two ways:

1- To choose the individuals with the highest fitness for reproduction. “Only the strong survive."

2- To assign a probability that a particular individual will be selected for either reproduction or crossover. This depend on our choice, because it allows for more diversity. Some weak individuals may contain branches of code which are strong.

The fitness function is determined subjectively. For example, we could include the depth of the tree as a potential quality we wish to control, and therefore we could develop a fitness function which takes this into account.

System components development for proposed model

I- Parameters on which this GA depends:

These are the number of generations (MAXGEN), population size (POPSIZE), probability of crossover (PXOVER), and probability of mutation (PMU).

There is no clear indication as to how large a population should be. If the population is too large, there may be difficulty in storing the data, but if the population is too small, there may not be enough string for good crossovers. In our experiment, a population consists of 50 members, i.e., POPSIZE = 50. In this case, PXOVER = 0.2, PMU = 0.2, and MAXGEN = 5000.

II- Chromosome representation:

An important issue in applying a GA is to design an appropriate chromosome representation of solutions of the problem together with genetic operators. Traditional binary vectors used to represent the chromosome are not effective in many highly non-linear physical problems. Since the proposed problem is highly non-linear, to overcome this difficulty, a real-number representation is used. In this representation, each chromosome V_i is a string of genes $G_{i\,j,}$ where genes $G_{i\,j}$ denote the decision variables p_{0N}, p_{1N} and p_{2N} tip1 and N and i denote the number of items and number of cycles, respectively. Since real number representation is used here, the value of each chromosome is the
actual value of the decision variable.

III- Initial population:

To initialize the population, we first determine the independent and dependent variables and then their boundaries.

All genes corresponding to all the independent variables are generated randomly between its boundaries and dependent variables are generated by different conditions.

IV- Evaluation:

The evaluation function plays the same role in the GA as that which the environment plays in natural evolution.

For this problem, the evaluation function is EVAL(Vi) = objective function value.

V- Selection:

Before the selection process, all chromosomes Vi are arranged in descending order according to their eval(Vi) and the roulette wheel selection process is applied on them POPSIZE times. Each time, a single chromosome is selected for the new population in the following way:

(a) Calculate the fitness value eval(Vi) for each chromosome VI.

(b) Find the total fitness of the population F= $\sum_{i=1}^{popsize} eval(v_i)$

(c) Calculate the probability of selection, pi = eval(υi)/F for each chromosomeVI .

(d) Calculate the cumulative probability qi for each chromosome VI: $q_i = \sum_{j=1}^{i} P_j$

(e) Generate a random real number r in (0, 1).

(f) If r < q1 then the first chromosome is V1; otherwise select the i^{th} chromosome

VI $(2 \leq i \leq \text{POPSIZE})$ such that $q_{i-1} < r \leq q_i$.

(g) Repeat steps (e) and (f) POPSIZE times and obtain POPSIZE copies of chromosomes.

By this process, better chromosomes may be selected several times depending upon the generated random numbers.

VI- Crossover operation:

The exploration and exploitation of the solution space is made possible by exchanging genetic information of the current chromosomes. Crossover operates on two parent solutions at a time and generates offspring solutions by recombining both parent solution features. After selection of chromosomes for the new population, the crossover operation is applied. Here, the whole arithmetic crossover operation is used. It is done in the following way:

(a) Firstly, we generate a random real number, r in (0, 1).

(b) Secondly, we select two chromosomes V_k and V_l randomly among population for crossover if r < PXOVR.

(c) Then two offspring $V_{0\,k}$ and $V_{0\,l}$ are produced as follows:

$$V_k' = c * V_k + (1-c) * V_1$$

$$V_1' = c * V_1 + (1-c) * V_k \quad \text{where } c \in [0, 1].$$

(d) Repeat the steps (a), (b) and (c) POPSIZE/2 times.

VII- Mutation operation:

Mutation operation is used to prevent the search process from converging to local optima rapidly. Unlike crossover, it is applied to a single chromosome Vi . Here, the mutation operation is defined as follows:

a) Firstly, we generate a random real number r in (0, 1).
b) Secondly, we select a chromosome Vi randomly from population if r < PMU.
c) Thirdly, we select a particular gene Gi j among the decision variables of the selected chromosome VI randomly.
d) Then the new gene corresponding to Gi j due to mutation is produced in the following way:
if the selected gene corresponds to the decision variable p_{0N}, p_{1N} and p_{2N}
if (RAND()$P_{kN} = P_{kN}$ + (U Bu − P_{kN})rnd()1−gen/MAXGENS
else $P_{kN} = P_{kN}$ − (P_{kN} − L Bu) rnd ()1−gen/MAXGENS Where N ∈ (0, 1, 2).
If the selected gene corresponds to the decision variable ti_{N1} then ti_N1 = iT/n + Rand Val (0, T/n);
e) Repeat the steps (a), (b), (c) and (d) POPSIZE times.

VIII- Termination:

If number of iterations is less than or equal to MAXGEN then the process continues; otherwise it terminates.

4. COMPUTATIONS

Let N = 2 n_1 = 2 and m = 1. Hence $P_N(\zeta) = P_{0N} + P_{1N}$ where $T_0(\zeta) = 1$ and $T1(\zeta) = \xi$. Now optimize the model (18) under the system of constrained (19, 20) via the GA technique to get the optimum value of p_{0N} and p_{1N}.

Min $J = J_1 + J_2 + n\,S_p$
Subject to

$$\sum_{N=1}^{2}\sum_{i=1}^{2} [\,(t_{iN1} - t_{i-1})\,(\sum_{j=0}^{2} b_{2j}(P_{0N} + P_{1N}\xi_j) + C_{pN} + n\,s_p] \le M$$

$$\sum_{N=1}^{2}\sum_{i=1}^{2} [\,(t_{iN1} - t_{i-1})\,(\sum_{j=0}^{k} b_{kj} \sum_{k=0}^{m} P_{kN} T_k(\xi_j))\, C_{pN} + n\,s_p\,] \le B_T$$

Input data

P = 2 T = 20 units $\tilde{s}_n$= (1.0, 1.2, 1.5)$

D01	D11	$\tilde{h}$ 1	h1	θ_1	$\tilde{C}$p1	C p1	a_1
1.94	0.5	(0.9,1.0 ,1.5) $	1.1 $	0.06	(1.1,1.5,1.9) $	1.6$	3.8
D02	D12	$\tilde{h}$ 2	h2	θ_2	$\tilde{C}$p2	C p2	a_2
2.70	0.2	(0.8, 1.3,2) $	1.35 $	0.05	(1.5,1.8,2.3) $	1.85$	2.8

Table-2: Input Data

Results

ℓ	P01	P11	P21	P02	P12	P22	J$
1	15.10	1.88	2.43	12.98	6.13	0.21	785.16
2	7.91	2.01	3.56	6.01	3.58	1.82	1449.02
3	15.08	2.98	2.04	11.02	3.10	0.20	1319.07
4	14.70	3.10	3.96	14.83	2.26	3.71	1081.99
5	12.03	2.60	1.99	6.9	0.54	0.31	776.79
6	8.99	2.22	0.46	11.90	2.82	2.71	801.96
7	12.37	1.60	0.71	14.97	10.98	3.01	886.33
8	15.99	1.27	0.69	6.00	20.02	0.03	897.62
9	11.06	6.02	3.71	9.16	1.98	0.20	919.05
10	16.28	2.21	3.05	12.92	5.85	0.25	932.41

Table-3: Results

The **o**ptimal productions and optimum cost are respectively
p1(ξ) = 9.98 + 1.12, p2(ξ) = 9.32 + 0.72 and J = 784.32$.

The optimal results with optimum number of cycles are presented in Table 3.

5. SENSITIVITY ANALYSIS

Here, a study has been made on the objective function due to the percentage change of holding cost and production cost. Table 3 gives the values of the objective function J for different values of h_N and C_{PN}, p = 1, 2. The percentage change of these values is shown with respect to the values used in the previous example with h_1, Cp_1, h_2,Cp_2 and the minimum objective value J = 776.79$. Table 4 shows that if hp and C_{pN} are increased/decreased by +5%, +10%, −5% and −10%, the values of the objective function change by 1.39% and 1.113%, 1.26% and 0%,1.83% and −1.99% and −2.14% and −2.57% respectively.

The result shows that the production cost is more sensitive with respect to the holding cost.

Change in hN(%)	Change in CpN (%)	Change in J (%)
10	0	1.39
0	9	1.26
5	0	1.11
0	7	0
−5	0	−1.83
0	−5	−1.99
−10	0	−2.14
0	−9	−2.57

Table-4: Sensitivity analysis

6. CONCLUSIONS

For the given data, it is observed from Table 3 that the total cost is minimum for the model in cycle 5 and it is maximum for cycle 2.

Most of the optimal control production inventory problems are analyzed by considering the whole finite time horizon as a single cycle. But represent multi-cycles result indicates that the above consideration does not correspond to minimum cost. Here, the optimum minimum cost (776.79$) is due to four (4) cycles within the given time horizon and this cost is much lower than the cost (1449.02 $) with the whole time horizon as a single cycle.

REFERENCES

1. M.D.S. Aliyu, A.A Andhani, Multi-item-multi-plant inventory control of production systems with shortages/ backorders, International Journal of Systems Science 30 (1999) 533–539.
2. Z.T. Balkhi, On a finite horizon production lot size inventory model for deteriorating items and optimal solution, European Journal of Operational Research 132 (2001) 210–223.
3. Z.T. Balkhi, An optimal solution of a general lot inventory model with deteriorated and imperfect products, taking into account inflation and time value of money, International Journal of System Science 35 (2004) 87–96.
4. M. Bendaya, A. Rauof, On the constraint multi-item single period inventory problem, International Journal of Production Management 13 (1993) 104–112.
5. L. Benkherouf, A. Bomenir, L. Aggoun, A diffusion inventory model for deteriorating items, Applied Mathematics and Computation 138 (2003) 21–39.
6. S.E. El-gendi, Chebyshev solution of differential, integral and integro-differential equations, Computer Journal (1969) 282–287.
7. T.M. El-Gindy, H.M. El-Hawary, M.S. Salim, M. El-Kady, A Chebyshev approximation for solving optimal control problems, Computers & Mathematics with Applications 29 (1995) 35–45.
8. L. Fox, I.B. Parker, Chebyshev Polynomials in Numerical Analysis, University Press, Oxford, 1972.
9. S.K. Goyal, B.C. Giri, Recent trends modelling of deteriorating inventory, European Journal of Operational Research 134 (2001) 1–16.
10. M.A. Hariga, L. Benkherouf, Optimal and heuristic inventory replenishment models for deteriorating items with exponential time-varying demand, European Journal of Operational Research 79 (1994) 123–137.
11. F. Harris, Operations and Cost, in: Factory Management Series, A.W. Shaw co., Chicago, 1915.
12. L. Li, K.K. Lai, A fuzzy approach to the multi-objective transportation problem, Computers and Operations Research 27 (2000) 43–57.
13. K. Maity, M. Maiti, Production inventory system for deteriorating multi-item with inventory-dependent dynamic demands under inflation and discounting, Tamsui Oxford Journal of Management Sciences 21 (2005) 1–18.
14. K. Maity, M. Maiti, Numerical approach of multi-objective optimal control problem in imprecise environment, Fuzzy Optimization and Decision Making 4 (2005) 313–330.

15. K. Maity, M. Maiti, Inventory of deteriorating complementary and substitute items with stock dependent demand, American Journal of Mathematical and Management Science 25 (2005) 83–96.
16. Montevechi B. Arnaldo José, Pinho F. Alexandre, Salomon A. Valério, "Triangular Fuzzy Numbers Utilization in Flowshop Sequence Algorithm", Escola Federal de Engenharia de Itajubá , (1999) , Brazil .
17. E. Naddor, Inventory Systems, Wiley, New York, 1996.
18. G. Padmanabhan, P. Vrat, Analysis of multi-systems under resource constraint. A nonlinear Goal programming approach, Engineering Cost and Production Management 13 (1990) 104–112.
19. S. Sana, S.K. Goyal, K.S. Chaudhuri, A production-inventory model for a deteriorating item with trended demand and shortages, European Journal of Operational Research 157 (2004) 357–371.
20. B.M. Worell, M.A. Hall, The analysis of inventory control model using polynomial geometric programming, International Journal of Production Research 20 (1982) 657–667.
21. Y.Wu. Zhou, H.S. Lau, S.L. Yang, A new variable production scheduling strategy for deteriorating items with time-varying demand and partial lost sale, Computers and Operations research 30 (2003) 1753–1776.

Source of support: Nil, Conflict of interest: None Declared.