Annals of **F**uzzy **M**athematics and **I**nformatics
Volume 12, No. 2, (August 2016), pp. 265–278
ISSN: 2093–9310 (print version)
ISSN: 2287–6235 (electronic version)
http://www.afmi.or.kr

@FMI

A multiobjective multi-item multilevel inventory-distribution system with imprecise cost via genetic algorithm

Nirmal Kumar Mahapatra

Received 3 June 2015; Revised 24 January 2016; Accepted 1 March 2016

Abstract. A multi-item inventory-distribution system via multi-warehouses and -retailers is considered in fuzzy-stochastic environment. Here lead-time demands of the items at different stations are considered as normally distributed with known mean and variance. On the other hand ordering cost, holding cost and retail-spaces are considered as fuzzy in nature. Here the model has been formulated as a multiobjective chance constraint-programming problem in fuzzy environment. The objectives are (i) minimization of expected annual stock-out at all stations and (ii) minimization of inventory related total cost of the system. The problem is then transferred to a multiobjective decision making (MODM) problem under crisp constraints. A multi objective genetic algorithm (MOGA) has been developed and implemented to solve the above MODM problem. The model has been illustrated by a numerical example and a set of compromise solution including optimum and near optimum ones has been presented.

2010 AMS Classification: 90B05, 90B30, 90C29, 90C30

Keywords: Inventory-distribution system, Multiobjective, Genetic algorithm, Chance constraint, Fuzzy-stochastic inventory.

Corresponding Author: Nirmal Kumar Mahapatra(nirmal_hridoy@yahoo.co.in)

1. Introduction

Since the development of EOQ model by Harris[10], extensive research work has been done in different areas of inventory control problems(Hadley and Whitin[9], Naddor[20], Tersine[29] etc). Now-a-days, due to the globalization of market with introduction of multinationals in the business, status of inventory and production problems faced by the industrial / business houses have been changed a lot. It is observed that multinationals store their product in a place of a country where maximum financial benefits in terms of rent, taxes etc. are available. Then products are distributed to different corners of the country / neighbouring countries via intermediate store houses. So, the interests of both multinationals (may be considered as wholesalers) and their franchisees (may be considered as retailers) have been coupled together and so objective of both the parties as well as the customer service are to be considered simultaneously. Hence, these considerations normally lead to the formulation of multiobjective decision making (MODM) inventory control problems.

Over the past two decades, extensive research work has been done to deal with more than one objective in inventory management problem. Bookbinder and Chen[3] developed a non-linear mixed integer programming model with two objectives for the warehouse–retailer system under deterministic demand. The objectives in this model are minimization of annual inventory and transportation costs. They also considered two probabilistic models with customers service as another objective. Ritha and Jeyakumari[25] developed a multi level inventory management decisions with transportation cost consideration in fuzzy environment. Noura *et. al* [22] solved fuzzy multiple objective minimum cost flow problem using credibility approach. Roy and Maiti[26] formulated an inventory problem of deteriorating items with two constraints, namely, storage space constraint and total average cost constraint and two objectives, namely, maximizing total average profit and minimizing total waste cost in fuzzy environment. Mahapatra and Maiti[17] developed a production - inventory model for a deteriorating item with imprecise preparation time for production in a finite time horizon.

Mahapatra[16] formulated a Multi-objective Inventory Model of Deteriorating Items with Some Constraints in an Intuitionistic Fuzzy Environment. Niknamfar[21] has developed a multi-objective production-distribution planning based on vendor-managed inventory strategy in a supply chain.

The major drawback of most of the above models is that they considered resources are crisp in nature or resource constraints deterministic. In real life, it is not so. During controlling period of inventory, the resource constraints may be possibilistic in nature i.e. it may happen that the constraints on resources will satisfy in almost all cases except possibly for a very few cases, where they may be allowed to violate. In stochastic environment for the solution of this type of problems, Mohan[18] proposed a 'here and now' approach- i.e. the chance constraints programming approach in which a minimum probability level for satisfying each of the constraint is specified. Similarly possibilistic constraints may also be defined (Liu and Iwamura[14]). Also, Bera *et. al.*[2] developed a multi-item mixture inventory model involving random lead time and demand with budget constraint and surprise function. Park[24], Eynan and Kropp[5], etc. also develop inventory models in stochastic environment. A Single Period Inventory Model with Discrete and Poisson Demand is also developed by Shah[27].

During last two decades, a set of evolutionary algorithms(EA) were suggested by different researchers to solve multiobjective decision making problems. Among these EAs, multiobjective genetic algorithms (MOGA) played a major role. An MOGA initially takes a population of feasible solutions for a problem, applies stochastic operations crossover and mutation on them for successive iterations and gives rise to a set of Pareto-optimal solutions. Then Decision Maker (DM) can make a choice for an appropriate / suitable solution among them according to his / her requirement.

Among these MOGAs, Fonseca and Fleming[6]'s MOGA, Deb *et. al.*[28]'s NSGA and Horn *et. al.*[11]'s NPGA enjoyed more attention. All these MOGA algorithms have a computational complexity of $O(MN^2)$ (where M is the number of objectives and N is the population size). But these techniques use a sharing function between the solutions for decision making. This sharing function is problem dependent and applicability of a method to a problem depends on this sharing function. Recently the authors of [4] proposed a MOGA depending on non-dominated sorting and sharing which does not require any sharing function and having a computational complexity of $O(MN^2)$. GA and MOGA have been applied to solve a few single and multiobjective inventory problems (e.g. Mondal and Maiti[19], Kumar*et. al.*[12], Lin and Song[13], Yousefi and Sadjadi[31], Arabzada*et. al.*[1], Guller*et. al.*[8] etc).

In this paper, an inventory distribution system is considered for multi-items in fuzzy-stochastic environment (probabilistic demand, fuzzy inventory related costs), where items are stocked at a central warehouse and distributed to retailers via intermediate warehouses. Finally the products are distributed to customers via retailers. Hence, the demand of a warehouse are determined by the sum of the demands of the retailers / warehouses under the corresponding warehouse. In case of probabilistic demand, since lead-time is the usual period of concern, attention is focused on the distribution of demand during the lead-time. Here ordering cost, holding cost and retail-spaces at different stations are considered as fuzzy in nature. Space constraints for different retailers is assumed to be possibilistic with some predefined confidence levels. Our objectives are to minimize inventory related total annual cost as well as to improve customer service. Customer service is improved by minimizing the stock-out occasions at different stations. So, the model has been formulated as a multiobjective fuzzy chance constraint programming problem. The objectives are (i) minimization of total expected annual stock-out at all stations and (ii) minimization of inventory related total cost of the system. Following Liu and Iwamura[15] the above multiobjective chance constraint programming problem is transferred to equivalent multiobjective crisp constraint problem. Then a multiobjective genetic algorithm (MOGA) has been developed (following Deb *et. al.*[4])and implemented to solve the above MODM problem. The model has been illustrated by a numerical example and a set of compromise solution including optimum and near optimum ones via MOGA has been presented.

2. Preliminaries

2.1. **Possibility in fuzzy environment:** Let R represents the set of real numbers. Then any fuzzy subset $\tilde{A}$ of R with membership function $\mu_{\tilde{A}} : R \to [0, 1]$, is called a fuzzy number. Let $\tilde{A}$ and $\tilde{B}$ be two fuzzy quantities with membership functions $\mu_{\tilde{A}}$ and $\mu_{\tilde{B}}$ respectively. Then according to Liu and Iwamura[14],

$$\text{Pos } (\tilde{A} \star \tilde{B}) = \sup\{\min(\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(y)), x, y \in R, x \star y\}, \tag{2.1}$$

where the abbreviation Pos represent possibility and $\star$ is any one of the relations $>, <, =, \leq, \geq$. Analogously if $\tilde{B}$ is a crisp number say b, then

$$\text{Pos } (\tilde{A} \star \tilde{B}) = \sup\{\mu_{\tilde{A}}(x), x \in R, x \star b\}. \tag{2.2}$$

If $\tilde{A}, \tilde{B}$ and $\tilde{C} = f(\tilde{A}, \tilde{B})$ are given, where $f : R \times R \to R$ is a binary operation, then membership function $\mu_{\tilde{C}}$ of $\tilde{C}$ is defined as

(2.3) $\mu_{\tilde{C}}(z) = \sup \{\min (\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(y)), x, y \in R \text{ and } z = f(x, y)\}, \forall z \in R.$

Let $\tilde{A} = (a_1, a_2, a_3, a_4)$ be a trapezoidal fuzzy number with membership function

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{x-a_1}{a_2-a_1} & \text{for } a_1 \leq x \leq a_2 \\ 1 & \text{for } a_2 \leq x \leq a_3 \\ \frac{a_4-x}{a_4-a_3} & \text{for } a_3 \leq x \leq a_4 \\ 0 & \text{otherwise} \end{cases} \tag{2.4}$$

and $\tilde{C} = (c_1, c_2)$ is a fuzzy number with membership function $\mu_{\tilde{C}}$ given by

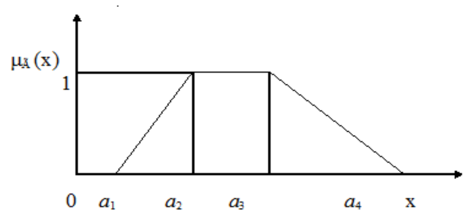

Figure - 1 : Trapezoidal fuzzy number

$$\mu_{\tilde{C}}(x) = \begin{cases} 1 & \text{for } 0 \leq x \leq c_1 \\ \frac{c_2-x}{c_2-c_1} & \text{for } c_1 \leq x \leq c_2 \\ 0 & \text{otherwise.} \end{cases} \tag{2.5}$$

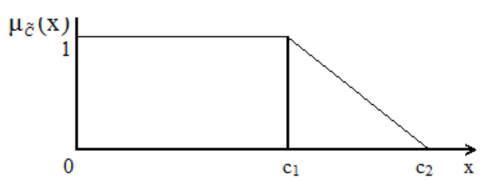

Figure - 2 : Graphical representation of a fuzzy number

Then, using the above definitions, following lemmas can easily be derived:

Lemma 2.1.

$$Pos(\tilde{A} \leq b) \geq \alpha \textit{ iff } \frac{b - a_1}{a_2 - a_1} \geq \alpha. \tag{2.6}$$

Lemma 2.2.

$$Pos(\tilde{C} \geq b) \geq \alpha \textit{ iff } \frac{c_2 - b}{c_2 - c_1} \geq \alpha. \tag{2.7}$$

2.2. **Fuzzy chance constraint programming:** A chance constrained programming with fuzzy parameters occurring in both constraints and objectives may be written as (Liu and Iwamura[15])

$$\text{(2.8)} \quad \begin{aligned} &\text{Min F} \\ &\text{subject to} \quad \text{Pos } \{\xi/f(x,\xi) \leq F\} \geq \beta \\ &\qquad\qquad\quad \text{Pos } \{\xi/g_j(x,\xi) \leq 0\} \geq \alpha_j, j = 1, 2, \cdots, n, \end{aligned}$$

where x is a decision vector, ξ is a vector of fuzzy parameters, $f(x,\xi)$ is the objective function, $g_i(x,\xi)$ are constraint functions and β, α_j are predetermined confidence level for objective function and the constraints respectively.

3. Model description

A multi-item inventory-distribution system via multi-warehouses and -retailers is considered. Here lead-time demands of the items at different stations are considered as normally distributed with known mean and variance. On the other hand ordering cost, holding cost and retail-spaces are considered as fuzzy in nature. The pictorial representation of the whole process can be depicted in following diagram. Here $i-$th warehouse at $j-$th level is denoted as WH_{ij}.

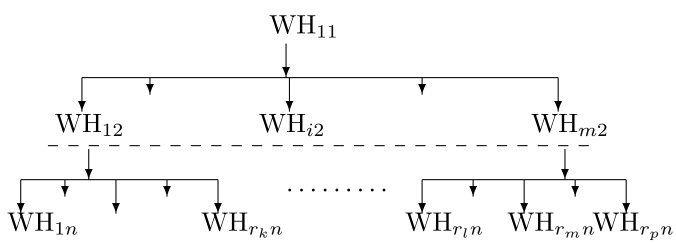

Figure-3: An n-level inventory-distribution system

3.1. **Assumptions and Notations.** A joint warehouses-retailers multi-item inventory lot size model is developed under the following notations and assumptions.

(i) A multi-echelon (n echelon) multi-item inventory distribution system is considered.

(ii) $i-$th warehouse at $j-$th level is denoted as WH_{ij}.

(iii) W_{ij} denotes capacity of WH_{ij}.

(iv) leaf warehouses, i.e., $n-$th level warehouses are assumed as retailers.

Following notations for $k-$th item at WH_{ij} are used for the present model.

D_{ijk} = Annual demand.

Q_{ijk} = Ordering quantity.

$\tilde{h}_{ijk}$ = Holding cost per unit.

$\tilde{A}_{ijk}$ = Ordering cost.

v_k = Space require to store one unit of $k-$th item.

L_{ijk} = Fixed replenishment lead time.

μ_{ijk} = Mean lead time demand.

σ^2_{ijk} = Variance of lead time demand.

f_{ijk} = Safety factor.
SS_{ijk} = Safety stock ($SS_{ijk} = f_{ijk}\sigma_{ijk}$).
s_{ijk} = Reorder point ($s_{ijk} = \mu_{ijk} + SS_{ijk}$).
P_{ijk} = Probability of stock out in any cycle.
Y = Sum of expected number of stock out occasions per year for all the stations at all the stations.
$\tilde{Z}$ = Sum of expected annual ordering cost and the average holding costs over the warehouses and retailers. The symbol tilde ( ) is used to represent a fuzzy parameter in the above expression.

4. Mathematical formulation

Since μ_{ijk} and σ_{ijk} are the mean and standard deviation of lead time demand of $k-$th item at WH$_{ij}$, so probability of stock out of $k-$th item at WH$_{ij}$ in any cycle (following approximation due to Page[23] and Tocher[30] is

$$(4.1)\quad P_{ijk} = \int_{S_{ijk}}^{\infty} \frac{1}{\sqrt{2\pi}\sigma_{ijk}} e^{-\frac{(x-\mu_{ijk})^2}{2\sigma_{ijk}^2}} dx = \int_{f_{ijk}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}} du \approx \frac{1}{1+e^{(2y_{ijk})}},$$

where $y_{ijk} = a_1 f_{ijk}(1 + a_2 f_{ijk}^2)$, with $a_1 = \sqrt{\frac{2}{\pi}}, a_2 = 0.044715$.
So expected number of annual stock out of $k-$th item at WH$_{ij}$ is

$$(4.2)\quad P_{ijk}\frac{D_{ijk}}{Q_{ijk}}$$

which implies that the expected number of stock out occasions of all the items considering all the stations is,

$$(4.3)\quad \text{Y} = \sum_i \sum_j \sum_k P_{ijk}\frac{D_{ijk}}{Q_{ijk}}.$$

The sum of expected annual ordering costs and average holding costs at all the stations is,

$$\tilde{Z} = \sum_i \sum_j \sum_k \left\{ \tilde{A}_{ijk}\frac{D_{ijk}}{Q_{ijk}} + \tilde{h}_{ijk}(\frac{1}{2}Q_{ijk} + f_{ijk}\sigma_{ijk}) \right\}.$$

Since, ordering cost and holding cost at different stations are assumed to be fuzzy in nature, the symbol tilde (˜) is used to represent a fuzzy parameter in the above expression.

Space constraints at different retailers are assumed to be possibilistic with some predefined confidence levels, so for $jk-$th retailer,

$$(4.4)\quad \text{Pos}\left\{ \sum_k v_k Q_{njk} \le \tilde{W}_{nj} \right\} \ge \alpha_{nj}, \forall j$$

Therefore the proposed model is

$$(4.5)\quad \text{Min } (\text{Y}, \tilde{Z})$$

subject to the space constraints given in equation(4.4). This problem can be reduced to a multiobjective fuzzy chance constraint programming problem as below.

$$\begin{aligned} (4.6) \qquad & \text{Min (Y, F)} \\ \text{subject to} \quad & \text{Pos}\ \{\tilde{Z} \le F\} \ge \alpha \\ & \text{Pos}\ \left\{ \sum_k v_k Q_{njk} \le \tilde{W}_{nj} \right\} \ge \alpha_{nj}, \forall j. \end{aligned}$$

5. Multiobjective genetic algorithm

Genetic Algorithm (GA) was introduced by John Holland[7] in the early 1970′s to solve decision making problems using the processes of natural evolution. Because of its generality it has been successfully applied to different branches of optimization problems, for its several advantages over conventional optimization methods. There are several algorithms to deal with multiobjective optimization problems using Genetic Algorithms. These algorithms are commonly known as multiobjective Genetic Algorithm (MOGA). An MOGA is developed to solve the proposed model. Different steps of this algorithm are given below:

Step 1: Generate initial population P_1 of size N.
Step 2: $i \leftarrow 1$ [i represent the number of current generation.]
Step 3: Select solution from P_i for crossover.
Step 4: Made crossover on selected solution to get child set C_1.
Step 5: Select solution from P_i for mutation.
Step 6: Made mutation on selected solution to get solution set C_2.
Step 7: Set $P_i' = P_i \bigcup C_1 \bigcup C_2$.
Step 8: Partition P_i' into subsets $F_1, F_2, \cdots, F_k$, such that each subset contains non-dominated solutions of P_i' and every solutions of F_i dominates every solutions of F_{i+1} for $i = 1, 2, \cdots, k-1$.
Step 9: Select maximum integer l, such that number of solutions in the set

$$F_1 \bigcup F_2 \bigcup \cdots \bigcup F_l \le N.$$

Step 10: Set $P_{i+1} = F_1 \bigcup F_2 \bigcup \cdots \bigcup F_l$.
Step 11: Sort F_{l+1} in decreasing order by crowding distance.
Step 12: Set M = number of solutions in P_{i+1}.
Step 13: Select first $N - M$ solutions from set F_{l+1}.
Step 14: Insert these solution in solution set P_{i+1}.
Step 15: Set $i \leftarrow i + 1$.
Step 16: If termination condition does not hold, goto step 3.
Step 17: Output P_i.
Step 18: End.

To implement the above multiobjective Genetic Algorithm (MOGA) for the proposed model, the following basic components are considered. (a) Parameters of MOGA, (b) Chromosome representation, (c) Initialization, (d) Genetic operators (crossover and mutation), (e) Crowding distance and (f) Non-dominated sorting. These components are given below:

(a) Parameters: Firstly, we set the different parameters on which this GA depends. All these are the number of generation (**MAXGEN**), population size (**POPSIZE**), probability of crossover (**PXOVER**), probability of mutation (**PMU**). There is no clear indication as to how large should a population 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, $POPSIZE = 100$, $PXOVER = 0.3$, $PMU = 0.2$, $MAXGEN = 5000$.

(b) Chromosome representation: Since the proposed problem is non-linear, a real - number representation of the chromosome is used. In this representation, each chromosome V_i is a string of n number of genes G_{ij}, (i=1,2,...POPSIZE and j=1,2,...,n) where these n number of genes respectively denote the number of decision variables.

(c) Initial population generations: For each chromosome V_i, every gene G_{ij} which represents an independent variable, is randomly generated between its boundary (LB_j, UB_j) where LB_j and UB_j are the lower and upper bounds of that variable and the gene G_{ij} which are the dependent variables, are generated from different conditions, until V_i is feasible,$i = 1, 2, ..., POPSIZE$.

(d) 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 new population, the crossover operation is applied. Here, the whole arithmetic crossover operation is used. It is done as following way:

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

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

(3) Then two offsprings V_k' and V_l' are produced as follows :

$$\begin{aligned} V_k' &= c * V_k + (1 - c) * V_l \\ V_l' &= c * V_l + (1 - c) * V_k, \quad \text{where } c \in [0, 1] \end{aligned}$$

(4) Repeat the steps (1),(2) and (3) **POPSIZE/2** number of times.

(e) 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 V_i. Here, the uniform mutation operation is used, which is defined as follows:

$$G_{ij}^{mut}\text{=random number from the range } (LB_j, UB_j),$$

where LB_j is the lower and UB_j is the upper boundary to the corresponding gene.

(f) Crowding distance: Crowding distance of a solution is measured using the following rule:

Step 1: Sort the population set according to every objective function values in ascending order of magnitude.

Step 2: For each objective function, the boundary solutions are assigned an infinite distance value. All other intermediate solutions are assigned a distance value equal to the absolute normalized difference in the function values of two adjacent solutions. This calculation is continued with other objective functions.

Step 3: The overall crowding distance value is calculated as the sum of the individual distance values corresponding to each objective.

Each objective function is normalized before calculating the crowding distance. Following algorithm is used for this purpose.

set k = number of solutions in F
for each k {
  set $F[k]_{distance} = 0$}
for each m {
  sort F, in ascending order of magnitude of m-th objective
  set $F[1]_{distance} = F[m]_{distance} = M$ where M is a large number
  for $i = 2$ to $k - 1$ {
  $F[i]_{distance} = F[i]_{distance} + (F[i+1]_m - F[i-1]_m)/(f_m^{max} - f_m^{min})$ }}

Here, $F[i]_m$ refers to the m-th objective function value of $F[i]$. f_m^{max} and f_m^{min} are the maximum and minimum values of the m-th objective function.

(g) Non-Dominated Sorting of a Population:

Here, for each solution the following two entities are calculated– (i) domination count n_p, the number of solutions which dominate the solution p, and (ii) S_p, a set of solutions that the solution p dominates. All solutions in the first non-dominated front will have their domination count as zero. Now, for each solution p with $n_p = 0$, we visit each member(q) of its set S_p and reduce its domination count by one. In doing so, if for any member q the domination count becomes zero, we put it in a separate list Q. These members belong to the second non-dominated front. Now, the above procedure is continued with each member of Q and the third front is identified. This process continues until all fronts are identified. The algorithm for non-dominated sorting approach is given below.

for each $p \in P$ {
set $S_p = \phi$ (where ϕ is a null set)
set $n_p = 0$
for each $q \in P$ {
  if (p dominates q) {
    $S_p = S_p \cup \{q\}$}
  elseif (q dominates p) {
    $n_p = n_p + 1$}}
if $n_p = 0$ {
  $p_{rank} = 1$
  $F_1 = F_1 \cup \{p\}$}}
set i=1
while $F_i \neq \phi$ {
  set $Q = \phi$

for each $p \in F_i$ {
  for each $q \in S_p$ {
    $n_q = n_q - 1$
    if $n_q = 0$ {
      $q_{rank} = i + 1$
      $Q = Q \cup \{q\}$ }}}
i=i+1
$F_i = Q$}

6. Numerical illustration

For numerical illustration, we consider a three level ($i.e. n = 3$) inventory distribution system as depicted in the following diagram.

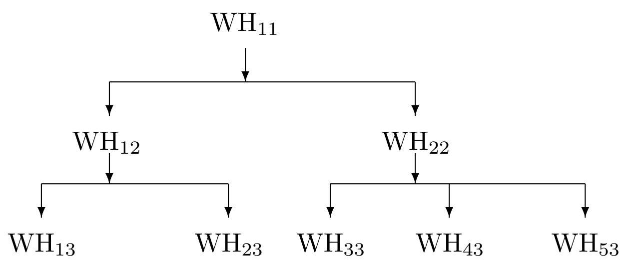

Figure-4: A 3-level inventory distribution system

Holding ($\tilde{h}_{ijk}$) and ordering costs of $k-$th item at WH$_{ij}$ are respectively assumed as trapezoidal fuzzy numbers $(h_{ijk1}, h_{ijk2}, h_{ijk3}, h_{ijk4})$ and $(A_{ijk1}, A_{ijk2}, A_{ijk3}, A_{ijk4})$. Then $\tilde{Z} = (Z_1, Z_2, Z_3, Z_4)$, where

$$Z_l = \sum_i \sum_j \sum_k \left\{ A_{ijkl} \frac{D_{ijk}}{Q_{ijk}} + h_{ijkl}(\frac{1}{2} Q_{ijk} + f_{ijk}\sigma_{ijk}) \right\}, (l = 1, 2, 3, 4).$$

We also assume $\tilde{W_{3j}} = (W_{3j1}, W_{3j2})$, graphical representation of whose membership function is similar to Figure-2.

According to above assumptions and using Lemmas 2.1 and 2.2, the problem (4.6) reduces to the following multiobjective optimization problem in crisp environment.

$$\begin{aligned} & \text{Min (Y, F)} \\ \text{subject to} \quad & \frac{F - Z_1}{Z_2 - Z_1} \geq \alpha \\ & \frac{W_{3j2} - \sum_k v_k Q_{ijk}}{W_{3j2} - W_{3j1}} \geq \alpha_{3j}, \forall j. \end{aligned} \tag{6.1}$$

To illustrate the above problem the following parametric values are considered.

Table-1: Input data for the model

St.	*item*	h_{ijk1}	h_{ijk2}	h_{ijk3}	h_{ijk4}	A_{ijk1}	A_{ijk2}	A_{ijk3}	A_{ijk4}	*AD*	*VLTD*	*Cap.*
WH_{13}	1	0.80	0.90	0.95	1.00	10	12	15	16	1000	15	150
	2	1.00	1.05	1.11	1.15	9	10	12	14	1150	20	160
WH_{23}	1	0.90	0.95	1.00	1.10	13	15	17	18	1200	17	155
	2	0.80	0.85	1.00	1.05	11	13	14	15	1100	15	165
WH_{33}	1	0.85	0.90	0.95	1.00	11	13	15	17	1100	20	160
	2	0.90	0.95	1.00	1.10	9	11	12	13	1000	17	170
WH_{43}	1	0.95	1.00	1.05	1.11	9	11	14	16	1200	16	150
	2	0.70	0.80	0.90	1.00	11	13	14	15	1100	18	160
WH_{53}	1	0.80	0.85	0.95	1.00	11	12	14	16	1300	18	155
	2	0.85	0.90	1.00	1.15	10	12	14	15	900	16	165
WH_{12}	1	0.45	0.50	0.55	0.60	20	23	25	26	−−	35	−−
	2	0.40	0.50	0.60	0.65	22	24	25	27	−−	40	−−
WH_{22}	1	0.40	0.45	0.55	0.70	22	25	28	30	−−	40	−−
	2	0.45	0.50	0.60	0.70	20	22	28	30	−−	35	−−
WH_{11}	1	0.20	0.25	0.35	0.40	50	55	65	70	−−	80	−−
	2	0.25	0.30	0.40	0.45	52	56	60	65	−−	90	−−

[St.=Station, AD=Annual Demand, VLTD=Variance of Lead Time Demand, Cap.=Capacity]

Also, let $v_1 = 0.50$ unit, $v_2 = 0.45$ unit, $\alpha = 0.9, \alpha_{3j} = 0.9, (j = 1, 2, \cdots, 5)$. For the above parametric values, a set of Pareto-optimal solutions is obtained via MOGA and some of these are presented in Table-2. A graphical representation of Pareto-optimal solutions are also presented in Figure-5.

Table-2: A set of Pareto-optimal Solutions of the Model

item *k*	Q_{13k} / f_{13k}	Q_{23k} / f_{23k}	Q_{33k} / f_{33k}	Q_{43k} / f_{43k}	Q_{53k} / f_{53k}	Q_{12k} / f_{12k}	Q_{22k} / f_{22k}	Q_{11k} / f_{11k}	Y	F
1	334.04	345.30	372.98	394.69	330.59	855.01	802.54	1633.41	0.0031	5388.21
	3.6034	3.9028	4.0978	3.6572	3.7458	3.7900	3.4570	3.8400		
2	336.76	282.72	312.65	376.84	336.48	677.72	646.04	1591.36		
	3.4912	4.2370	3.8161	4.0984	3.8659	4.0634	4.0133	3.9443		
1	280.29	312.78	323.27	387.77	181.12	852.21	733.64	1651.15	0.0429	5015.30
	2.9487	2.9648	2.8940	3.1684	3.3854	3.3225	3.2558	2.9680		
2	310.68	274.18	250.83	293.18	328.26	689.76	638.09	1573.31		
	4.0837	3.2883	3.1697	3.9043	3.4801	3.4258	2.8125	3.1785		
1	311.02	323.21	346.24	369.40	299.46	852.82	791.90	1612.71	0.0115	5160.16
	3.3147	3.4659	3.5624	3.2874	3.4790	3.4883	3.3775	3.3091		
2	316.15	277.31	272.30	334.22	333.25	678.97	644.08	1480.50		
	3.6213	3.6664	3.5279	3.9468	3.6148	3.6391	3.3200	3.4938		
1	300.08	313.74	332.81	356.19	284.73	852.41	788.40	1595.21	0.0239	5058.65
	3.1301	3.2629	3.3569	3.0929	3.3277	3.3500	3.3212	3.1227		
2	307.70	274.76	255.08	313.14	331.76	679.14	643.76	1432.87		
	3.6179	3.3923	3.3905	3.8986	3.5054	3.4652	3.0473	3.3103		
1	303.58	316.70	337.09	360.20	289.33	852.52	789.66	1602.86	0.0189	5090.53
	3.1910	3.3272	3.4201	3.1541	3.3746	3.3914	3.3395	3.1814		
2	310.30	275.56	260.38	320.02	332.25	679.04	643.96	1456.46		
	3.6156	3.5061	3.4349	3.9118	3.5393	3.5190	3.1313	3.3675		
1	316.04	327.16	352.26	374.50	305.68	852.94	794.23	1616.68	0.0085	5203.62
	3.4059	3.5560	3.6492	3.3708	3.5369	3.5432	3.4048	3.3860		
2	319.53	278.39	279.28	344.06	333.98	678.70	644.22	1499.31		
	3.6152	3.7888	3.5898	3.9768	3.6603	3.7100	3.4331	3.5705		
1	286.14	311.79	324.83	375.74	280.99	852.21	752.87	1617.75	0.0380	5017.87
	2.9600	3.0764	3.0240	3.1201	3.3575	3.3242	3.2635	2.9965		
2	308.73	274.15	250.92	294.20	329.07	686.68	639.66	1506.24		
	3.9775	3.2931	3.2068	3.9015	3.4808	3.4264	2.8321	3.2086		

When stock-out is plotted against the annual inventory cost, the following pictorial representation is obtained.

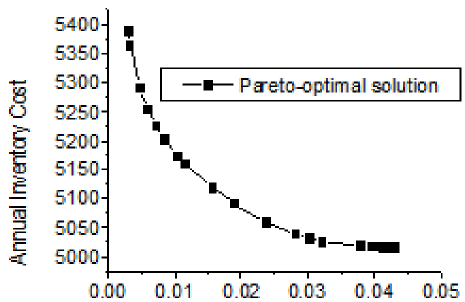

Figure - 5 : Expected stockout

7. Discussion

It is observed from Table-2 that when annual inventory cost (F) decreases, expected number of stock out in a year increases, which agrees with reality. When safety factors are determined then reorder point at any station can easily be determined. Figure-5 represents optimum policy curve for two objectives. Any point on the curve gives an optimum policy for decision maker(DM), points off the curve are suboptimal but can be improved by moving back to the curve. In this way, DM can improve his / her policy according to requirement.

8. Conclusion

A realistic multi-item inventory distribution problem under a supply chain system in fuzzy stochastic environment is presented and solved via multiobjective genetic algorithm. As a unique optimal solution might not always be a solution for the DM, the solution needed by the DM is multiple solution subject to both objective and resource constraints under different criteria preferred by DM. In this paper, a neighbouring domain of optimal solutions has been presented and it is to be noted that every solution in this neighbouring domain is acceptable. It is up to DM to choose a particular near optimal solution for implementation considering the presently prevailing condition of his / her factory / concern. Moreover, though only three level distribution system has been considered, the present supply chain problem can be extended to any number of levels for distribution.

References

[1] S. Mohammad Arabzada, Mazaher Ghorbanib and Reza Tavakkoli-Moghaddamc, An evolutionary algorithm for a new multi-objective location-inventory model in a distribution network with transportation modes and third-party logistics providers, International Journal of Production Research 53 (4) (2015) 1038–1050.

[2] U. K. Bera, M. Rong, N. K. Mahapatra and M. Maiti, A multi-item mixture inventory model involving random lead time and demand with budget constraint and surprise function, Applied Mathematical Modelling 33 (2009) 4337–4344.

[3] J. H. Bookbinder and V. Y. X. Chen, Multicriteria Trade-offs in a Warehouse Retailer System, Journal of Operations Research Society 43 (7) (1992) 707–720.

[4] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, A Fast and Elitist Multiobjective GEnetic Algorithm: NSGA-II, IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION 6 (2) (2002) 182–197.

[5] A. Eynan and D. H. Kropp, Effective and simple EOQ-like solutions for stochastic demand periodic review systems, Europian Journal of Operation Research 180 (2007) 1135 –1143.

[6] C. M. Fonseca and P. J. Fleming, Genetic Algorithms for Multiobjective Optimization, Formulation, Discussion and Generalization, Proceedings of the Fifth International Conference on Genetic Algorithm, S. Forrest, Ed. San Mateo, CA: Morgan Kauffman (1993) 416–423.

[7] D. E. Goldberg, Genetic Algorithms: Search, optimization and machine learning, Massachusetts, Addison Wesley 1989.

[8] M. Guller, Y. Uygun and B. Noche, Simulation-based optimization for a capacitated multi-echelon production-inventory system, Journal of Simulation 9 (2015) 325–336.

[9] G. Hadely and T. M. Whitin, Analysis of inventory systems, Prentice Hall, Englewood, Cliffs, NJ 1963.

[10] F. Harris, Operations and Cost, Factory Management Series, Chicago: A. W. Shaw Co. (1915) 48–52.

[11] J. Horn, N. Nafploitis and D. E. Goldberg, A niched Pareto Genetic Algorithm for Multiobjective Optimization,in Proceedings of the First IEEE conference on Evolutionary Computation, Z. Michalewicz, Ed.Piscataway, NJ: IEEE Press (1994) 82–87.

[12] Tarun Kumar, S. R. Singh and C. B. Gupta, Genetic Algorithm Based Multi Product and Multi Agent Inventory Optimization in Supply Chain Management, International Journal of Modeling and Optimization 2 (6) (2012) 653–657.

[13] Lin Lin and Shiji Song, Hybrid NSGA-II algorithm on robust multi-objective inventory management problem, Transactions of the Institute of Measurement and Control 37 (7) (2015) 909–918.

[14] B. Liu and K. Iwamura, Chance constraint Programming with fuzzy parameters, Fuzzy Sets and Systems 94 (1998) 227–237.

[15] B. Liu and K. Iwamura, A note on chance constrained programming with fuzzy coefficients, Fuzzy Sets and Systems 100 (1998) 229–233.

[16] N. K. Mahapatra, Multi-objective Inventory Model of Deteriorating Items with Some Constraints in an Intuitionistic Fuzzy EnvironmentInternational Journal of Physical and Social Sciences 2 (9) (2012) 342–363.

[17] N. K. Mahapatra and M. Maiti,ProductionÔÇôinventory model for a deteriorating item with imprecise preparation time for production in a finite time horizon, Asia-Paci´$\frac{1}{4}$üc Journal of Operational Research 23 (2) (2006) 171–192.

[18] C. Mohan, Optimization in fuzzy-stochastic environment and its importance in present day industrial scenario, Proceedings on "Mathematics and its application in industry and business", Narosa Publishing House, India 2000.

[19] S. Mondal and M. Maiti, Multi-item fuzzy EOQ models using genetic algorithm, Computers & Industrial Engineering 44 (2002) 105–117.

[20] E. Naddor, Inventory systems, John Wiley, New York 1966.

[21] A. H. Niknamfar , Multi-objective production-distribution planning based on vendor-managed inventory strategy in a supply chain, Industrial Management and Data Systems 115 (6) (2015) 1086–1112.

[22] A. A. Noura, A. Payen and E. Poodineh, Credibility approach to solve fuzzy multiple objective minimum cost flow problem, Ann. Fuzzy Math. Inform. 1 (2) (2011) 221–235.

[23] E. Page, Approximations to the cumulative normal function and its inverse for use on a pocket calculator, Evol. Appl. Stat. 26 (1977) 75–76.

[24] C. Park, An analysis of the lead time demand distribution derivation in stochastic inventory systems, International Journal Production Economics 105 (2007) 263–272.

[25] W. Ritha and S. Rexlin Jeyakumari, Multi level inventory management decisions with transportation cost consideration in fuzzy environment, Ann. Fuzzy Math. Inform. 2 (2) (2011) 171–181.

[26] T. K. Roy and M. Maiti, Multiobjective inventory models of deteriorating items with some constraints in a fuzzy environment, Computers and Operations Research 25 (1998) 1085–1095.

[27] A. N. Shah, A Single Period Inventory Model with Discrete and Poisson Demand, International Journal Of Scientific Research 2 (7) (2013) 506–508.

[28] N. Srinivas and K. Deb, Multiobjective function optimization using nondominated sorting genetic algorithms,Evol. Comput. 2 (3) (1995) 221–248.

[29] R. J. Tersine, Principles of Inventory and Material Management, PTR Prentice Hall, New Jersey 1994.

[30] K. D. Tocher, The Art of Simulation, English University Press, London 1963.

[31] Ommolbanin Yotsefi and Seyed Jafar Sadjadi, Solving a new bi-objective joint replenishment inventory model with modified RAND and genetic algorithms, Turkish Journal of Electrical Engineering and Computer Sciences 22 (2014) 1338–1353.

Nirmal Kumar Mahapatra (nirmal_hridoy@yahoo.co.in)
Department of Mathematics, Panskura Banamali College, Panskura RS-721152, WB, India