An application of extended elitist non-dominated sorting Genetic Algorithm in multi-objective linear programming problem of tea industry with interval objectives

Asoke Kumar Bhunia[a], Amiya Biswas[a*] and Nabendu Sen[b]

[a]*Department of Mathematics, The University of Burdwan, Burdwan, India*
[b]*Department of Mathematics, Assam University, Silchar, India*

C H R O N I C L E	A B S T R A C T
	In this paper, we have modeled a decision making problem of a tea industry as a multi-objective optimization problem in interval environment. The goal of this problem is to maximize the overall profit as well as to minimize the total production cost subject to the given resource constraints depending on budget, storage space and allotted processing times in different machines. For this purpose, the problem has been formulated as a multi-objective integer linear programming problem with interval objectives. To solve the problem, we have proposed extended elitist non-dominated sorting genetic algorithm (ENSGA-II) for integer variables with interval fitness, crowded tournament selection, intermediate crossover, one neighborhood mutation and elitism. To develop this algorithm, we have proposed modified non-dominated sorting and crowding distance based on interval mathematics and interval order relations. Finally, to test the performance of the proposed algorithm, a numerical example has been solved.

1. Introduction

In most of the real-world decision making problems, it is not always feasible or desirable to reduce all the goals of an organization into single objective. To overcome this difficulty, the corresponding problem can be formulated as an optimization problem with multiple objectives of conflicting interest. This type of problem is known as multi-objective optimization problem. On the other hand, due to globalization of economy and competitive market situation, the executives of an industry are forced to minimize the system cost and also to maximize the profit simultaneously. For this purpose, they have to face some unpleasant situation due to uncertainty of several system parameters. The values of those parameters are imprecise. To solve the problems with such imprecise numbers, generally stochastic, fuzzy and fuzzy stochastic approaches are applied. In these approaches, problems are converted into deterministic/crisp problems. In stochastic approach, the parameters are assumed to be random variables with known probability distribution. In fuzzy approach, the

parameters, constraints and goals are considered as fuzzy set with known membership functions or fuzzy numbers. On the other hand, in fuzzy stochastic approach, some parameters are considered as fuzzy sets/ fuzzy numbers and others as random variables. However, to specify the appropriate membership function for fuzzy approach, type of fuzzy numbers, probability distribution for stochastic approach and both for fuzzy-stochastic approach is a formidable task to a decision maker. So, to overcome these difficulties for handling the imprecise numbers by different approaches, one may use interval number for representing the imprecise number as this representation is the most significant representation among others. As a result, the objective functions would be interval valued. The general form of multi-objective optimization problems with interval objectives can be written as

Optimize $\{A_1(x), A_2(x), ..., A_k(x)\}$
subject to $x \in S$
where $A_j(x) = [f_{jL}(x), f_{jR}(x)], j = 1, 2, ..., k$
and $S = \{x : g_i(x) \leq 0; i = 1, 2, ..., m\}$

Over the last few decades, several techniques have been developed for solving multi-objective optimization problem. The details about different methods/ techniques are available in the existing literature (Miettinen et al., 2004; Deb, 2000). Among these methods, non-dominated sorting genetic algorithm (NSGA) is very promising one. In all the methods except NSGA, the corresponding multi-objective optimization problem is converted to a single optimization problem by giving biasness/ preference/ weightage to one or more objectives. However, in NSGA, this type of preference is not given. In this area, the first pioneering work was done by Srinivas and Deb (1995). Thereafter, Deb et al. (2000) modified NSGA and proposed a computationally fast elitist genetic algorithm (called NSGA-II) based on non-dominated sorting. The proposed algorithm of Deb et al.(2000) is the first breakthrough work in this area. After Deb et al. (2000), several researchers applied this algorithm for solving their problems.

In Tea industry, there arise different types of decision making problems. The main objective of these problems is to determine the quantity of different grades of tea so that the overall profit is maximized or the overall cost is minimized or both. To the best of our knowledge, no work has been done in solving the decision making problems of tea industry considering multiple objectives. Deb (1999) first solved the transportation problem of tea industries of Barak Valley of Assam in India. Sinha and Sen (2011) formulated and solved a goal programming model for some industries. Recently, Sen (2012) extended the work of Sinha and Sen (2011) by considering several goals based on profit, production demand, use of processing machines. In all these studies, the values of the system parameters were precise i.e., the values of the parameters were considered in crisp environment. However, in reality, these parameters may not be precise due to uncertainty (mere precisely due to fluctuation of market, human error, improper storage and other unexpected factors relating to environment).

The objective of this paper is to model a decision making problem of a tea industry as a multi-objective integer linear programming problem with two interval valued conflicting objectives. The goal of the problem is to maximize the overall profit as well as to minimize the total production cost subject to the given resource constraints depending on budget, storage space and allotted times of different machines. To solve the corresponding problem, we have extended the existing algorithm NSGA-II for interval valued fitness (we call it Extended NSGA-II (ENSGA-II)). In this algorithm, we have proposed modified non-dominated sorting and crowding distance based on interval mathematics and interval order relations. Finally, to test the performance of the proposed algorithm, a numerical example has been solved.

2. Assumptions and Notations:

The following assumptions and notations have been used in the entire paper.

2.1. Assumptions

(i) The company produces more than one varieties of tea.
(ii) The demand of each variety of tea is deterministic and its value is a certain fraction of the total production of that variety.
(iii) The different varieties of tea are processed in different machines (say, m machines).
(iv) There is a particular amount of budget allocated for the total production.
(v) The produced different varieties of tea will be stored in a warehouse.

2.2. Notations

n	The number of varieties of tea grade
M	Number of machines in which different varieties of tea are processed
x_i	Produced quantity of i-th grade of tea, $i = 1, 2, ..., n$
p_i	Unit profit of i-th grade of tea
c_i	Cost of production per unit of i-th grade of tea
t_{ji}	Processing time for unit production of i-th grade of tea in j-th machine, j=1,2,…,m.
b_i	Proposed budget per unit for i-th grade of tea
A_i	Storage space of X_i ($X_i = 25x_i$)
l_i	Fraction of production of i-th grade of tea
B	Total budget allocation
T_j	Available time for j-th machine
P	Expected total production
E	The total expenditure (factory expenditure)
D	Expected total demand
A	Total space of warehouse of the tea estate
p_s	Population size
m_g	Maximum generation
p_c	Probability of crossover
p_m	Probability of mutation

3. Prerequisite

An interval number A is a closed connected subset of $\mathbb{R}$ denoted by $A = [a_L, a_R]$ and defined by $A = [a_L, a_R] = \{x : a_L \leq x \leq a_R, x \in \mathbb{R}\}$, where a_L and a_R are the left and right limits respectively. An interval number can also be expressed in terms of its center and radius as $A = \langle a_c, a_w \rangle$, where a_c and a_w are the center and the radius of the interval A respectively i.e., $a_c = (a_L + a_R)/2$ and $a_w = (a_R - a_L)/2$. Actually, every real number $x \in \mathbb{R}$ can be stated as an interval number $[x, x]$ with center x and radius zero.

3.1 Interval Arithmetic

According to Moore (2009), the definitions of addition, subtraction, multiplication, and division of interval numbers are as follows:

Definition: If $A = [a_L, a_R]$ and $B = [b_L, b_R]$

$A + B = [a_L, a_R] + [b_L, b_R] = [a_L + b_L, a_R + b_R]$

$A - B = [a_L, a_R] - [b_L, b_R] = [a_L - b_R, a_R - b_L]$

For any real number λ,

$$\lambda A = \lambda [a_L, a_R] = \begin{cases} [\lambda a_L, \lambda a_R] & if \ \lambda \geq 0 \\ [\lambda a_R, \lambda a_L] & if \ \lambda < 0, \end{cases}$$

$$A \times B = \left[\min\{a_L b_L, a_L b_R, a_R b_L, a_R b_R\}, \max\{a_L b_L, a_L b_R, a_R b_L, a_R b_R\} \right]$$

$$\frac{A}{B} = A \times \frac{1}{B} = [a_L, a_R] \times \left[\frac{1}{b_R}, \frac{1}{b_L} \right], \text{ provided } 0 \notin [b_L, b_R].$$

Definition: In the centre and radius form of interval, addition, subtraction and multiplication by a scalar of interval numbers are defined as follows:

Now if $A = \langle a_c, a_w \rangle$ and $B = \langle b_c, b_w \rangle$

$A + B = \langle a_c, a_w \rangle + \langle b_c, b_w \rangle = \langle a_c + b_c, a_w + b_w \rangle$

$A - B = \langle a_c, a_w \rangle - \langle b_c, b_w \rangle = \langle a_c - b_c, a_w + b_w \rangle$

For any real number λ,

$\lambda A = \lambda \langle a_c, a_w \rangle = \langle \lambda a_c, |\lambda| a_w \rangle$

3.2 Interval order relations

Interval order relation plays an important role in solving the decision making problems with interval objectives. Any two closed intervals $A = [a_L, a_R] = \langle a_c, a_w \rangle$ and $B = [b_L, b_R] = \langle b_c, b_w \rangle$ may be of the following three types.
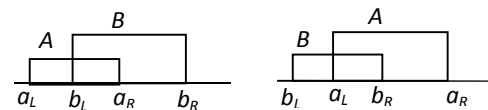
Type I: Non-overlapping intervals [Fig. 1(a)]
Type II: Partially overlapping intervals [Fig. 1(b)]
Type III: Fully overlapping intervals [Fig. 1(c)]
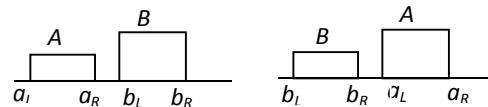
Fig. 1(a) Type – I intervals

Fig. 1(b) Type – II intervals
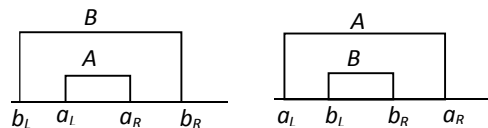
Fig. 1(c) Type – III intervals

Over the last few decades, several researchers proposed the definitions of order relations between two interval numbers. Recently, Sahoo et al. (2012) proposed the same by rectifying the drawbacks of the existing definitions. Very recently, Bhunia and Samanta (2013) proposed alternative definitions of order relations for maximization and minimization problems separately. Their definitions are as follows:

Definition: The order relation $\geq^{\max}$ between two intervals $A = [a_L, a_R] = \langle a_c, a_w \rangle$ and $B = [b_L, b_R] = \langle b_c, b_w \rangle$ for maximization problem is as follows:

$$A \geq^{\max} B \Leftrightarrow \begin{cases} a_c > b_c & if\ a_c \neq b_c \\ a_w \leq b_w & if\ a_c = b_c \end{cases}$$

and $A >^{\max} B \Leftrightarrow A \geq^{\max} B$ and $A \neq B$.

Definition: The order relation $\leq^{\min}$ between two intervals $A = [a_L, a_R] = \langle a_c, a_w \rangle$ and $B = [b_L, b_R] = \langle b_c, b_w \rangle$ for minimization problem is as follows:

$$A \leq^{\min} B \Leftrightarrow \begin{cases} a_c < b_c & if\ a_c \neq b_c \\ a_w \leq b_w & if\ a_c = b_c \end{cases}$$

and $A <^{\min} B \Leftrightarrow A \leq^{\min} B$ and $A \neq B$.

Also the intervals $A = [a_L, a_R]$ and $B = [b_L, b_R]$ are equal if and only if $a_L = b_L$ and $a_R = b_R$ [Moore (1979)].

3.3 Interval Metric

Definition: Let M be any non empty set. A function $d : M \times M \to \mathbb{IR}$, is said to an interval metric if it satisfies the following properties:

(i) Reflexivity: $d(X,X)_c = 0 \in d(X,X)$

(ii) Triangular inequality $d(X,Y) \leq^{\min} d(X,Z) + d(Z,Y)$

(iii) Symmetry: $d(X,Y) = d(Y,X)$

(iv) Indiscernible identity: If $d(X,Y) = d(X,X)$ then $X = Y$.

Definition: Let X and $Y \in \mathbb{IR}$. An interval distance between X and Y is denoted by $d_I(X,Y)$ and is defined by

$$d_I(X,Y) = \tilde{\lceil} X - Y \tilde{\rceil} = \tilde{\lceil} \langle x_c, x_w \rangle - \langle y_c, y_w \rangle \tilde{\rceil} = \tilde{\lceil} \langle x_c - y_c, x_w + y_w \rangle \tilde{\rceil} = \langle |x_c - y_c|, x_w + y_w \rangle$$

Definition: Let $X = (X_1, X_2, ..., X_k)$ and $Y = (Y_1, Y_2, ..., Y_k) \in \mathbb{IR}^k$. An interval distance between X and Y is denoted by $d_I(X,Y)$ and is defined by

$$d_I(X,Y) = \sum_{i=1}^{k} \tilde{\lceil} X_i - Y_i \tilde{\rceil}$$

4 Mathematical formulation of the problem

Let us assume that a tea company produces different varieties (say, *n*) of tea according to the size of the leaves. Before production, these grades of tea undergo through different processes in different machines (say,*m*). It is observed that these grades of tea take different times in the respective machines and the total production of all varieties of tea depends upon the factory expenditure and the demands on different markets around the locality of the said tea estate and outside also. The objective of the problem is to maximize the overall profit and also to minimize the total cost subject to the given resource constraints.

According to the assumptions and notations, the mathematical formulation of the optimization problem is as follows:

Problem-1:

Maximize $z_1(x) = \sum_i p_i x_i$

Minimize $z_2(x) = \sum_i c_i x_i$

subject to the constraints

$g_j(x) = \sum_i t_{ji} x_i - T_j \leq 0, j = 1, 2, ..., m$

$g_{m+1}(x) = \sum_i b_i x_i - B \leq 0$

$g_{m+2}(x) = \sum_i A_i x_i - A \leq 0$

$x_i \geq 0$ and are integers, $i = 1, 2, ..., n$

Now if the profit as well as production cost per unit of each grade of tea are imprecise and these impreciseness are represented by interval numbers, then the total profit and total production cost will be interval valued. In this situation, the general form of Problem-1 can be written as follows:

Problem-2:

Maximize$[z_{1L}(x), z_{1R}(x)] = \sum_i [p_{iL}, p_{iR}] x_i$

Minimize$[z_{2L}(x), z_{2R}(x)] = \sum_i [c_{iL}, c_{iR}] x_i$

subject to the constraints

$g_j(x) = \sum_i t_{ji} x_i - T_j \leq 0, j = 1, 2, ..., m$

$g_{m+1}(x) = \sum_i b_i x_i - B \leq 0$

$g_{m+2}(x) = \sum_i A_i x_i - A \leq 0$

$x_i \geq 0$ and are integers, $i = 1, 2, ..., n$

5. Solution Procedure

As the problem-2 is constrained optimization problems, so we can solve the same by penalty function technique. In this technique, the constrained optimization problem is converted into unconstrained optimization problem. Here we have used the Big-M penalty technique (Gupta et. al.(2009)). Hence the unconstrained optimization problem corresponding to the Problem 2 is as follows

Problem 3:

Maximize$[\bar{z}_{1L}(x), \bar{z}_{1R}(x)] = \begin{cases} [z_{1L}(x), z_{1R}(x)] \text{ when } x \in S \\ [-M, -M] \text{ when } x \notin S \end{cases}$

Minimize$[\bar{z}_{2L}(x), \bar{z}_{2R}(x)] = \begin{cases} [z_{2L}(x), z_{2R}(x)] \text{ when } x \in S \\ [M, M] \text{ when } x \notin S \end{cases}$

where $S = \{x : g_k(x) \leq 0, k = 1, 2, ..., m+2, x_i \text{ is integer}, i = 1, 2, ..., n\}$

This is a multi-objective integer linear programming problem with two conflicting objectives. Here both the objectives are interval valued. To solve the problem, we have extended the existing elitist non-dominated sorting genetic algorithm (NSGA-II) due to Deb et al. (2000) for interval valued objectives.

This algorithm is dependent on the following criteria:
  (i)   Chromosome representation and initialization
  (ii)   Evaluation of fitness function
  (iii)   Modified non-dominated sorting
  (iv)   Modified crowding distance
  (v)   Modified crowded Tournament selection
  (vi)   Genetic operators (Crossover, mutation and elitism)

5.1 Chromosome representation and initialization

The aim of tea industry is to determine the quantity of different grades of tea by maximizing the overall profit as well as by minimizing the total production cost. Here the quantity of tea produced for each grade of tea is considered to be an integer. So all the genes of a chromosome are integer valued. To represent such chromosome, here we have used real coding representation. In that case, the value of each gene (i.e., solution component) is chosen randomly between the lower and upper bounds of the corresponding decision variable. There are several procedures for selecting a random integer number. In this work, we have used the following for selecting an integer random number.
 A random integer number x between a and b can be generated by either $x = a + g$ or $x = a - g$ where g is a random integer between 1 and $|a - b|$.

5.2 Evaluation of fitness function

In GA, fitness function plays an important role which is same as natural evolution process in the biological and physical environments. After initialization of chromosomes, it is very important to check how relatively good they are. For this purpose, first of all, the fitness values of all the chromosomes are to be computed. In this problem, there are two conflicting objectives corresponding to each chromosome, viz. to maximize the overall profit and to minimize the total production cost.

5.3 Non-dominated sorting

In the proposed problem, both the objectives of the problem are interval valued. So, in order to sort a population of size p_s according to the level of non-domination, each solution must be compared with each other solution in the population to check whether is it dominated or not? This process is continued to select the members of the first non-dominated front from all population members. In order to find the members for the subsequent fronts, the members of the first front are temporarily rejected and the procedure is repeated until all the solutions are exhausted. For example, if we consider two chromosomes $\underset{\sim}{a} = (a_1, a_2, ..., a_n)$ and $\underset{\sim}{b} = (b_1, b_2, ..., b_n)$, then

(i) for minimization problem, $\underset{\sim}{a}$ is said to dominate $\underset{\sim}{b}$ if $f_i(\underset{\sim}{a}) \leq^{\min} f_i(\underset{\sim}{b}) \ \forall i = 1, 2, ..., n$ and $f_i(\underset{\sim}{a}) <^{\min} f_i(\underset{\sim}{b})$ for atleast one i.

(ii) For maximization problem, $\underset{\sim}{a}$ is said to dominate $\underset{\sim}{b}$ if $f_i(\underset{\sim}{a}) \geq^{\max} f_i(\underset{\sim}{b}) \ \forall i = 1, 2, ..., n$ and $f_i(\underset{\sim}{a}) >^{\max} f_i(\underset{\sim}{b})$ for atleast one i.

Otherwise these two chromosomes are said to be non-dominated.

5.4 Crowding distance

Crowding distance is an important factor in the selection process of NSGA-II. Mainly, this distance is measured for the solutions of same front. For this purpose, all the chromosomes of a particular front are sorted first according to each objective function value in descending order (in ascending order for minimization problem) of its magnitude. Then for each objective function, the boundary solutions are assigned to an infinite distance value. All other intermediate solutions are assigned to a distance which is equal to the absolute normalized difference of the function values of two adjacent solutions. The crowding distance of intermediate solution is defined by:

$$dI_j^m = dI_j^m + \frac{f_m^{(I_{j+1}^m)} - f_m^{(I_{j-1}^m)}}{f_m^{max} - f_m^{min}}, m = 1, 2, j = 2, 3, ..., L-1$$

where I_j denotes the solution index of the j-th member in the sorted list, L be the number of solutions, I_1 and I_L be the lowest and highest objective function values, f_m^{max} and f_m^{min} be the population-maximum and population-minimum values of the m-th (m=1,2) objective function. This calculation is continued with other objective functions. Finally, the overall crowding distance is computed as the sum of the individual distance values corresponding to each objective.

5.5 Crowded Tournament Selection

This operation is dependent on the attributes of each solution (i.e., solution i)

    (i)   A non-domination rank r_i in the population.

    (ii)   A local crowding distance (d_i) of the population.

By tournament selection, solution i will be selected from solutions i and j if any of the following conditions is true:

    (i) If the rank of solution i is less than that of solution j i.e., $r_i < r_j$.

    (ii) If both the solution have same rank but solution i has a better crowding distance than solution j, i.e., if $r_i = r_j$ and $d_i > d_j$.

The first condition ensures that the selected solution lies in a better non-dominated front. The second condition resolves the tie between two solutions which are in the same non-dominated front by deciding on their crowding distances.

5.6 Crossover and mutation

After the selection process, other two main genetic operators viz., crossover and mutation are applied to the resulting chromosomes those which have survived. The goal of this operation is to exchange the information between randomly selected parent chromosomes (individuals) by combining their features. In this work, intermediate crossover has been used. The different steps are as follows:

Step 1: Find the integral value of $p_c * p_s$ and store it in M_1.

Step 2: Select two chromosomes v_k and v_l randomly from the population.

Step 3: Generate a random integer between 0 and $|v_{kj} - v_{lj}|, j = 1, 2, ..., n$, where v_{kj} and v_{lj} be the j-th gene/ components of the individuals v_k and v_l respectively.

Step 4: Compute the components $\bar{v}_{kj}$ and $\bar{v}_{lj}$ (j=1,2,…,n) of two offspring by

either $\bar{v}_{kj} = v_{kj} - g$ and $\bar{v}_{lj} = v_{lj} + g$ if $v_{kj} > v_{lj}$

or, $\bar{v}_{kj} = v_{kj} + g$ and $\bar{v}_{lj} = v_{lj} - g$,otherwise

Step 5: Repeat Steps 2-4 for $\dfrac{M_1}{2}$ times.

The goal of mutation operation is to introduce the random variation into the population. Mainly, this operation is used for fine tuning of the system. Also, it helps to regain the information lost in earlier generation. In this work, one-neighbourhood mutation has been used. The details of one-neighborhood mutation is available in Bhunia et al. (2010).

6. Numerical illustration

To illustrate the problem numerically and also to test the performance of the proposed algorithm, we have considered a bi-objective decision making problem of a tea industry of Barak valley, Assam, India who produces 5 grades of tea, viz., Pekoe dust, Orange fanning, Broken Pekoe, Broken Orange Pekoe (BOP), BOP (small). The values of different parameters are presented in Table 1. The objective of this problem is to determine the number of units of different grades of tea (x_1 units of Pekoe dust, x_2 units of Orange fanning, x_3 units Broken pekoe, x_4 units of BOP (small), x_5 units of BOP) to be produced per month by optimizing two conflicting objectives, viz. maximization of overall profit and minimization of total production cost. The corresponding problem has been formulated as interval bi-objective integer LPP and solved by the proposed algorithm. Considering the fixed values of profit and production cost per unit, the corresponding problem has also been solved.

The proposed algorithm is coded in C programming language and run in a LINUX environment. The computational work has been done on a PC with Intel core-2 duo processor with 1 GB RAM. Due to stochastic nature of the algorithm, in each case, 50 independent runs have been performed to compute the best found values of production quantity of different grades of tea along with the overall profit and the total production cost. In each case, we have considered only two solutions, one with largest profit and other with least production cost. Among 50 runs, 10 different Pareto-optimal solutions have been obtained in cases of the problems in interval and crisp environments respectively. Those results have been shown in Table 2 and 3. From the economical point of view, the manager of the tea industry may choose that solution where overall profit is largest. In this context, the acceptable solutions of the industry are as follows (from Tables 2 and 3):

For the problem with interval valued profit and production cost

$x_1 = 800, x_2 = 9, x_3 = 20, x_4 = 49, x_5 = 1006$

Overall profit (Rs.)=[47853.60,68754.60] and total production cost (Rs.)=[145237.00,176296.00] For the problem with fixed value of profit and production cost

$x_1 = 827, x_2 = 8, x_3 = 15, x_4 = 35, x_5 = 1275$

Overall profit=Rs. 54724.20 and total production cost=Rs. 163570.00

In the simulation process, the following values of different parameters of the problems and the proposed algorithm are used:

For interval valued objectives
$p_s = 300, m_g = 350, p_c = 0.8, p_m = 0.1$

For fixed objectives
$p_s = 400, m_g = 400, p_c = 0.8, p_m = 0.1$

Table 1
Values of different parameters

Types of grades of Tea(No. of units to be produced)	X_i in Rs.	t_{1i} (in Hr.)	t_{2i} (in Hr.)	$A_i = 25a_i$ (in m^3)	p_i (in Rs.) (Fixed)	$[p_{iL}, p_{iR}]$ in Rs.	c_i in Rs.	$[c_{iL}, c_{iR}]$ in Rs.
Pekoe dust (x_1)	118	0.95×10^{-3}	0.60×10^{-3}	0.45	28	[28,38]	94	[94,114]
Orange fanning (x_2)	112	0.85×10^{-3}	0.55×10^{-3}	0.65	15.4	[15.4,25.4]	89	[89,90]
Broken pekoe (x_3)	106	0.80×10^{-3}	0.50×10^{-3}	0.63	12	[12,22]	76	[76,90]
BOP(small)(x_4)	100	0.76×10^{-3}	0.48×10^{-3}	0.74	19	[19,30]	68	[68,82]
BOP(x_5)	98	0.72×10^{-3}	0.45×10^{-3}	0.86	24	[24,36]	64	[64,78]

$\chi = $ Rs. 2,30,000.00, $\beta_1 = 130$ hr., $\beta_2 = 125$ hr., $A = 1500$ m^3

Table 2
Pareto-optimal best found values of problem with interval profit and problem cost

Sl. No.	x_1	x_2	x_3	x_4	x_5	Overall Profit (Rs.)	Total production cost (Rs.)	No. of chromosomes in 1st front
1	800	42	26	45	1001	[48237.80, 69424.79]	[148038.00,179088.00]	437
	900	47	34	74	1097	[54065.80, 77853.79]	[166607.00, 201524.00]	
2	800	8	27	53	1005	[47974.20, 68967.20]	[145888.00, 177086.00]	431
	900	45	34	79	1094	[54058.00, 77845.00]	[166577.00, 201520.00]	
3	800	26	34	42	1001	[48030.40, 69104.40]	[147018.00, 178122.00]	451
	899	49	35	76	1093	[54022.60, 77804.60]	[166647.00, 201532.00]	
4	800	18	21	44	1001	[47789.20, 68675.20]	[145454.00, 176396.00]	439
	900	47	35	78	1092	[54033.80, 77815.79]	[166635.00, 201552.00]	
5	800	34	32	53	1003	[48386.60, 69665.60]	[148454.00, 179720.00]	421
	899	49	33	71	1100	[54071.60, 77862.60]	[166603.00, 201488.00]	
6	800	27	29	45	1001	[48042.80, 69109.79]	[146931.00, 178008.00]	437
	900	50	35	76	1090	[53994.00, 77760.00]	[166638.00, 201502.00]	
7	800	9	20	49	1006	[47853.60, 68754.60]	[145237.00, 176296.00]	438
	900	48	35	71	1098	[54060.20, 77847.20]	[166632.00, 201536.00]	
8	800	27	19	41	1000	[47822.80, 68733.79]	[145835.00, 176702.00]	426
	900	47	34	76	1095	[54055.80, 77841.79]	[166615.00, 201532.00]	
9	800	36	22	46	1005	[48212.40, 69358.40]	[147524.00, 178582.00]	435
	900	50	33	69	1100	[54077.00, 77866.00]	[166650.00, 201528.00]	
10	800	18	25	42	1003	[47847.20, 68775.20]	[145750.00, 176748.00]	426
	900	47	35	72	1098	[54063.80, 77851.79]	[166611.00, 201528.00]	

Table 3
Pareto optimal best found values of problem with fixed profit and production cost

Sl. No.	x_1	x_2	x_3	x_4	x_5	Overall Profit (Rs.)	Total Production Cost (Rs.)	No. of chromosomes in 1st front
1	827	8	15	35	1275	54724.20	163570.00	62
2	832	13	18	35	1275	54977.20	164713.00	65
3	833	12	18	35	1275	54989.80	164718.00	62
4	832	15	15	36	1275	54991.00	164731.00	62
5	833	14	15	35	1276	55008.60	164732.00	63

To test the convergence of the Pareto-optimal set, the following metric has been used

$$\Delta = \frac{d_f + d_l + \sum_{i=1}^{N-1} \left| d_i - \overline{d} \right|}{d_f + d_l + (N-1)\overline{d}}.$$

Here d_i is the Euclidean distance between consecutive solutions in the obtained non-dominated set of solutions and $\overline{d}$ is the average of the distances d_i (i=1,2,…,N-1), assuming that there are N solutions in the best non-dominated front. Also, d_f and d_l are the Euclidean distances between the extreme solutions and the boundary solutions. The lesser value of variance of Δ gives better convergence of the algorithm. The mean and variance of Δ has been calculated by performing 50 consecutive runs for both fixed and interval objectives, which are as follows:

Table 4
Mean and variance of Δ

	Mean	Variance
Fixed objective values	0.5162	0.0006
Interval objective values	1.8166	0.0006

6 Concluding Remarks

For the first time, we have formulated a decision making problem of a Tea industry as bi-objective linear programming problems. Considering interval valued profit and interval cost per unit. For solving this problem, for the first time, we have developed extended NSGA-II with the help of interval mathematics and interval order relations. For future research, other parameters of the problem, such as total budget, allotted times of different machines and allotted space for storage of packed tea, can also be considered as uncertain parameters. Furthermore, other meta-heuristic methods can be used to solve the problem.

Acknowledgement

References

Bhunia, A. K., & Samanta, S. S. (2014). A study of interval metric and its application in Multi-objective optimization with interval objective. *Computers & Industrial Engineering, 74, 169-178.*

Bhunia, A.K., Sahoo, L. & Kapoor, P.K. (2012). Genetic algorithm based multi-objective reliability optimization in interval environment. *Computers and Industrial Engineering*, 62, 152-160.

Bhunia, A.K., Sahoo, L. & Roy, D. (2010). Reliability stochastic optimization for a series system with interval component reliability via Genetic Algorithm. *Applied Mathematics and Computation*, 216(3), 929-939.

Deb, K., Agrawal, S., Pratap, A. & Meyarivan, T. (2000). A Fast Elitist Non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. *Proceedings of the Parallel Problem Solving from Nature VI Conference*, 16-20 September, Paris, France, 849-858.

Gupta, R.K., Bhunia A.K. & Roy, D. (2009). A GA based penalty function technique for solving constrained redundancy allocation problem of series system with interval valued reliabilities of components. *Journal of Computational and Applied Mathematics*, 232(2), 275-284.

Hansen, E. & Walster, G.W. (2004). Global optimization using interval analysis. Marcel Dekker Inc, New York.

Hu, B. Q. & Wang, S. (2006). A novel approach in uncertain programming Part I: New arithmetic and order relation for interval numbers. *Journal of Industrial and Management Optimization*, 2(4), 351-371.

Karmakar, S., Mahato, S.K. & Bhunia, A.K. (2009). Interval oriented multi-section techniques for global optimization. *Journal of Computation and Applied Mathematics*, 224(2), 476-491.

Miettinen, K. (2004). *Nonlinear multi-objective optimization*. Kluwer Academic Publishers.

Nahar, S., Sahni, S. & Shragowitz, E. (1986). Simulated annealing and combinatorial optimization. *Proceedings of the 23rd ACM/IEEE Design Automation Conference* , 293-299.

Moore. R.E. (1979). *Method and Applications of Interval Analysis*, SIAM, Philadelphia.

Sen, N. (2012). A note on Goal Programming Approach to Tea Industry of Barak Valley of Assam. *American Journal of Mathematics and Statistics,* 5,108-113.

Sinha, B. & Sen, N. (2011). Goal Programming approach to tea industry of Barak Valley of Assam. *Applied Mathematical Sciences*, 5(29),1409-1419.

Srinivas, N. & Deb, K. (1995). Multi-Objective function optimization using non-dominated sorting genetic algorithms. *Evolutionary Computation*, 2(3), 221–248.