



# Journal of Applied Sciences

ISSN 1812-5654

**science**  
alert

**ANSI***net*  
an open access publisher  
<http://ansinet.com>

## A Methodology for Optimizing Statistical Multi-Response Problems Using Genetic Local Search Algorithm Through Fuzzy Goal Programming

<sup>1</sup>M. Amiri, <sup>2</sup>N. Karimi and <sup>2</sup>S.F. Jamshidi

<sup>1</sup>Faculty of Management and Accounting, Valiasr Avenue, Allame Tabatabaei University, Tehran, Iran

<sup>2</sup>Faculty of Industrial Engineering, Islamic Azad University, Branch of Qazvin, Qazvin, Iran

---

**Abstract:** This study presents a methodology for solving multi-response optimization problems. Since goal programming method considers decision maker's comments objectively, it has special significance; but using this method in large and complex problems alone can't be, so effective, thus it would be a better idea to use a metaheuristic method. The proposed method is a combination of simulation approach, fuzzy goal programming, genetic algorithm and local search algorithm. This method will use firstly simulation to generate required inputs, secondly fuzzy goal programming to model the problem and finally genetic local search algorithm for problem optimization. At the end we will show the performance of this method by numerical example and designed experiments.

**Key words:** Statistical multi-response problem, fuzzy goal programming, genetic algorithm, local search algorithm

---

### INTRODUCTION

In multi objective decision making environments, one of general problem is to select a set of input conditions (the X's as independent variables) which results in a product with a desirable set of outputs (the Y's as response variables). In fact this problem is simultaneous optimization of the response variables, each of which depends upon a set of independent variables  $X_1, \dots, X_n$ . In this study we wish to select the levels of the independent variables in such a way that all the response variables become optimized.

For instance, in quality control environments maybe the goal is to find the levels of the input variables (quality characteristics) of the process so that the quality of the product (response) will have the desired characteristics. Moreover, in Response Surface Methodology (RSM), we adjust the levels of the input variables to the point that the set of outputs become optimized (Montgomery, 1997).

Multi-response optimization in the framework of RSM is an attempt to this end. Different optimization methods have been presented before but we present a hybrid method for optimizing statistical multi-response problems. The important characteristic of our method is that it uses simulation in generating problem's input data and performs a combination of genetic algorithm with local search for optimization of model in order to improve the solution.

Classical methods such as bounded objectives method or a combination of bounded objectives and lexicography methods are used in a multi-response optimization.

Loss function method was used by Pignatiello (1993), Winston (2003), Artiles-Leon (1996), Ross (1996) and Kapur and Cho (1996). The basis of this method is on Taguchi's loss function. The basic idea in this method was transforming multi objective problem to a single objective one.

Later Cheng *et al.* (2002) extended Zimmermann's method for optimizing statistical multi-response problems but the solution's length was the weakness of their method. Pasandideh modeled statistical multi-response optimization problems using desirability function and also four different GA methods by simulation method to solve it (Pasandideh and Niaki, 2006).

Kim and Rhee (2004) proposed a method based on the desirability function and GA. They applied this method to optimize a welding process.

Khoo and Chen (2001) combined genetic algorithm with RSM and also used genetic algorithm in three different scenarios for optimizing optimization parameter of one and two response.

Avello *et al.* (2004) used simulated annealing method for optimizing multi-response problems. In each stage of solution, they used a strategy for selection of basic

function, so in each stage a function would be chosen as a basic function and others as constraints. Ozselik and Erzurumlu (2005) presented an optimization method using RSM and GA to minimize the bend of slim layer of plastic.

This was an expecting model for capability of plastic bending using RSM by three level DOE. This model was used with GA to find optimum amount of process parameter.

To analyze several qualitative attributes simultaneously, Koksoy and Yalcinoz (2006) used RSM and compared its results with solution of GA method. Correia *et al.* (2005) offered a comparison between RSM and GA in optimization of welding process (GMAW). Suresh *et al.* (2002) presented second level model for the expecting roughness degree of steel parts. Their study tries to optimize roughness degree of the parts by GA. Oktem *et al.* (2005) offered a GA method using RSM in which RSM presents an effective model to determine the level of parameter and GA optimizes these levels.

The problem has been solved using a Pareto approach by assigning different weights to prioritize the objectives. However, when the problem becomes large or when the structure is complex, this GP approach is totally impractical. Chao-Fang and *et al.* (2007) offered a fuzzy goal programming by giving different priority to objectives by DM. Gen *et al.* (1993) used 0-1 fuzzy goal programming approach in order to solve optimization problems.

Several works have been done on genetic algorithm by applying goal programming. Mirrazavi *et al.* (2001) analyzed the use of genetic algorithms to solve Integer Goal Programming models in a general frame. Chen and Liu (1994) used genetic algorithm to solve a Weighted Goal Programming model.

**NOMENCLATURE**

The following notations will be used throughout this study:

- $U_j$  : Upper bound of objective function  $Z_j$
- $L_j$  : Lower bound of objective function  $Z_j$
- $\alpha_j$  : Access level to the optimum of objective function  $Z_j$
- $W_j$  : The importance or weight of objective function  $Z_j$  from the view point of the decision maker
- $\mu(Z_j)$  : The fuzzy membership function of objective function  $Z_j$
- $Z_j^*$  : The optimal value of objective function  $Z_j$

- $X_j^*$  : The optimal value of variable  $X_j$ .
- $\Delta_j$  : The tolerance of objective function  $Z_j$   $\Delta_j = U_j - L_j$

**DEFINITIONS**

**Definition 1:** The mathematical model of the multi objective problem is defined as following:

$$\begin{aligned} \text{Max } Z_j &= Y_j(X_1, X_2, \dots, X_n) \quad j = 1, 2, \dots, m \\ -1 &\leq X_i \leq 1 \quad i = 1, 2, \dots, n \end{aligned}$$

**Definition 2:** If in a multi objective problem with m objectives, each objective function is solved independently then we will have m independent optimal solutions. By replacing each optimal solution in the other objective functions, we will gain a lower and upper limit for each objective function. For the jth objective function, we solve the following problem separately ( $j = 1, 2, \dots, m$ ). Table 1 shows the results.

Where:

- $Z_{ij}$  = The values of the jth objective function in terms of optimal variables of the ith objective function problem
- $X_{ij}$  = The optimal value of variable  $X_j$  in the ith objective function

**Definition 3:** For any objective function, there is a fuzzy membership function. Figure 1 shows fuzzy membership function.

Where:

$$U_j = Z_j^* \quad L_j = \text{Min}_i(Z_{ij}) \quad \Delta_j = U_j - L_j$$

Table 1: The range of objective functions

	$Z_1$	$Z_2$	...	$Z_m$	$X_1$	$X_2$	...	$X_n$
Max ( $Z_1$ )	$Z_{11} = Z_1^*$	$Z_{12} = Z_2$	...	$Z_{1m} = Z_m$	$X_{11}$	$X_{12}$	...	$X_{1n}$
Max ( $Z_2$ )	$Z_{21} = Z_1$	$Z_{22} = Z_2^*$	...	$Z_{2m} = Z_m$	$X_{21}$	$X_{22}$	...	$X_{2n}$
Max ( $Z_m$ )	$Z_{m1} = Z_1$	$Z_{m2} = Z_2$	...	$Z_{mm} = Z_m^*$	$X_{m1}$	$X_{m2}$	...	$X_{mn}$

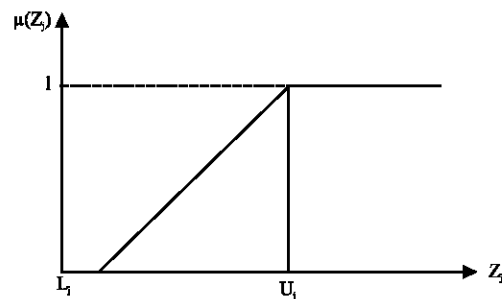


Fig. 1: One-sided fuzzy membership function of  $Z_j$  function

$$\mu(Z_j) = \begin{cases} 0 & Z_j < U_j - \Delta_j \\ \frac{Z_j - (U_j - \Delta_j)}{\Delta_j} & U_j - \Delta_j \leq Z_j < U_j \\ 1 & Z_j \geq U_j \end{cases}$$

**PROBLEM MODELING**

Here, in this study, we consider following assumptions:

All the factors, which make up the input of the problem, are the independent variables  $X_1, X_2, \dots, X_n$ .

The lower and the upper bounds of the independent variables are -1 and 1 where  $X_i$  is a coded variable in such a way that  $-1 \leq X_i \leq 1$ .

The output of the problem is the response variables given by  $Y_1, Y_2, \dots, Y_k$ . For every objective function  $Z_j$  we have:  $L_j \leq Z_j \leq U_j$ .

The one-sided or two-sided transformation for each response depends (on the nature of) the objective of the problem. The mathematical model of the problem becomes:

$$\begin{aligned} \text{Max: } & Z_j = Y_j(X_1, X_2, \dots, X_n) \quad j = 1, 2, \dots, m \\ \text{s.t:} & \\ & -1 \leq X_i \leq 1 \quad i = 1, 2, \dots, m \end{aligned}$$

**Theorem 1:** Consider the following response variables problem:

$$\begin{aligned} \text{Max: } & Z_j = Y_j(X_1, X_2, \dots, X_n) \quad j = 1, 2, \dots, m \\ \text{s.t:} & \\ & -1 \leq X_i \leq 1 \quad i = 1, 2, \dots, m \end{aligned}$$

Assuming an identical importance for the objectives and the identical access level to the optimal point of each objective, the decision maker's desirable solution will be found by solving the following mathematical programming model:

$$\begin{aligned} \text{Max : a} & \\ \text{s.t:} & \\ & \frac{Z_j}{\Delta_j} + n_j - p_j = \frac{U_j}{\Delta_j} \\ & a + n_j \leq 1 \\ & -1 \leq X_j \leq 1 \\ & a \in [0 \ 1] \\ & n_j \geq 0, p_j \geq 0 \quad j = 1, 2, \dots, m, \quad i = 1, \dots, n \end{aligned}$$

Where:

- $p_j$  : The function positive deviation  $Z_j/\Delta_j$
- $n_j$  : The function negative deviation  $Z_j/\Delta_j$
- $\alpha$  : Access level to the optimum of any objective function

**Proof:** In Fig. 1 we have :

$$\mu(Z_j) = \begin{cases} 0 & Z_j < U_j - \Delta_j \\ \frac{Z_j - (U_j - \Delta_j)}{\Delta_j} & U_j - \Delta_j \leq Z_j < U_j \\ 1 & U_j \leq Z_j \end{cases}$$

$$a = \text{Min} \mu(Z_j) \Rightarrow$$

Max : a

$$\text{a. } \alpha \leq \frac{Z_j}{\Delta_j} - \frac{U_j}{\Delta_j} + 1, \quad \frac{U_j}{\Delta_j} - 1 \leq \frac{Z_j}{\Delta_j} \leq \frac{U_j}{\Delta_j} \quad \text{for some } j$$

$$\text{b. } \frac{U_j}{\Delta_j} \leq \frac{Z_j}{\Delta_j} \leq \frac{U_j}{\Delta_j} + 1 \quad \text{for some } j$$

For the constraints of section a, suppose that:

$$\frac{Z_j}{\Delta_j} = \frac{U_j}{\Delta_j} - n_j \quad (n_j : \text{negative deviation})$$

Now we have:

$$a + n_j \leq 1, \quad \frac{Z_j}{\Delta_j} + n_j = \frac{U_j}{\Delta_j} \quad (1)$$

For the constraints of section b, suppose that:

$$\frac{Z_j}{\Delta_j} = p_j + \frac{U_j}{\Delta_j}$$

Now we have:

$$\frac{Z_j}{\Delta_j} - p_j = \frac{U_j}{\Delta_j} \quad (2)$$

Now we combine constraint (1) and (2) and finally:

$$\begin{aligned} \text{Max : } & \sum_{j=1}^m W_j \alpha_j \\ & \frac{Z_j}{\Delta_j} + n_j - p_j = \frac{U_j}{\Delta_j} \\ & a_j + n_j + p_j \leq 1 \\ & -1 \leq X_i \leq 1 \\ & a_j \in [0 \ 1] \\ & n_j \geq 0, p_j \geq 0, \quad i = 1, \dots, n \end{aligned}$$

**Corollary 1:** If the access level to the optimum state of each objective is different and the objectives are not identically important from the view point of decision maker, then the desirable solution of the decision maker will be obtained by solving the following mathematical model:

$$\begin{aligned} \text{Max: } & \sum_{j=1}^m W_j \alpha_j \\ & \frac{Z_j + n_j - p_j}{\Delta_j} = \frac{U_j}{\Delta_j} \\ & \alpha_j + n_j \leq 1 \\ & -1 \leq X_i \leq 1 \\ & \alpha_j \in [0 \ 1] \\ & n_j \geq 0, p_j \geq 0, \quad i = 1, \dots, n \end{aligned}$$

**Proof:** This is the result of theorem 1 with the following assumptions:

$$\alpha_j \leq \mu(Z_j)$$

**Corollary 2:** If the response variables have the best nominal value (two-sided transformation) and the objectives are not identically important from the view point of the decision maker and the access level to the optimum status of each objective is different then the decision maker's desirable solution will be obtained by solving the following mathematical model:

$$\begin{aligned} \text{Max: } & \sum_{j=1}^m W_j \alpha_j \\ & \frac{Z_j + n_j - p_j}{\Delta_j} = \frac{T_j}{\Delta_j} \\ & \alpha_j + n_j + p_j \leq 1 \\ & -1 \leq X_i \leq 1 \\ & \alpha_j \in [0 \ 1] \quad p_j \geq 0 \quad n_j \geq 0 \quad i = 1, \dots, n \end{aligned}$$

Where:

$T_j$  = Nominal value of objective function  $Z_j$

**Proof:** This is the result of theorem 1 with the following assumptions. Figure 2 shows the two-sided fuzzy membership function membership function.

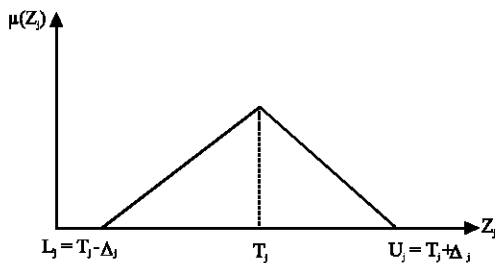


Fig. 2: Two-sided fuzzy membership function

$$\mu(Z_j) = \begin{cases} 0 & Z_j < T_j - \Delta_j \\ \frac{Z_j - (T_j - \Delta_j)}{\Delta_j} & T_j - \Delta_j \leq Z_j < T_j \\ \frac{T_j + \Delta_j - Z_j}{\Delta_j} & T \leq Z_j < T_j + \Delta_j \\ 0 & Z_j \geq T_j + \Delta_j \end{cases}$$

$$\alpha_j \leq \mu(Z_j)$$

**Corollary 3:** If there are k response variables on the more-the better (one-sided transformation) and m-k response variables on the better nominal value (two-sided transformation) and the goals are not identically important from the viewpoint of the decision maker, then the desired solution from the viewpoint of the decision maker will be obtained by the following mathematical model:

$$\begin{aligned} \text{Max: } & \sum_{j=1}^m W_j \alpha_j \\ & \frac{Z_j + n_j - p_j}{\Delta_j} = \frac{U_j}{\Delta_j} & j = 1, 2, \dots, k \\ & \frac{Z_j + n_j - p_j}{\Delta_j} = \frac{T_j}{\Delta_j} & j = k+1, k+2, \dots, m \\ & \alpha_j + n_j \leq 1 & j = 1, 2, \dots, k \\ & \alpha_j + n_j + p_j \leq 1 & j = k+1, k+2, \dots, m \\ & -1 \leq X_i \leq 1 & i = 1, 2, \dots, n \\ & \alpha_j \in [0 \ 1] & n_j \geq 0 \quad p_j \geq 0 \end{aligned}$$

The proof is obtained from Corollaries 1 and 2. Considering the proof of this theorem, we propose an algorithmic procedure for calculating the response surface.

## METHODOLOGY

**Step 1:** Define the following response variables problem:

$$\begin{aligned} \text{Max: } & Z_j = Y_j(X_1 \ X_2 \dots \ X_n) \quad j=1, 2, \dots, k \\ \text{Min: } & r_j = R_j(X_1 \ X_2 \dots \ X_n) \quad j=k+1 \ k+2, \dots, m \\ \text{s.t: } & -1 \leq X_i \leq 1 \quad i=1, 2, \dots, n \end{aligned}$$

**Step 2:** Change the response variables problem as following:

$$\begin{aligned} \text{Max: } & Z_j = Y_j(X_1 \ X_2 \dots \ X_n) \quad j=1, 2, \dots, m \\ \text{s.t: } & -1 \leq X_i \leq 1 \quad i=1, 2, \dots, n \end{aligned}$$

**Step 3:** Solve the problem for each objective function separately and obtain the solutions. Put the solutions in the objective functions and obtain

for each objective function two lower limit ( $L_j$ ) and upper limit ( $U_j$ ) as the best and the worst case and then obtain  $\Delta_j = U_j - L_j$

**Step 4:** Obtain the objective weights  $W_j$  from the decision maker and then solve the following mathematical programming model:

$$\begin{aligned} \text{Max: } & \sum_{j=1}^m W_j \alpha_j \\ & \frac{Z_j + n_j - P_j}{\Delta_j} = \frac{U_j}{\Delta_j} \\ & \alpha_j + n_j \leq 1 \\ & -1 \leq X_i \leq 1 \\ & \alpha_j \in [0 \ 1] \\ & p_j \geq 0, n_j \geq 0 \quad j=1, 2, \dots, m \quad i=1, \dots, n \end{aligned}$$

Now in order to obtain optimum or near optimum solution, we use a metaheuristic algorithm, a genetic local search, which searches for best solution in an extended space.

**Genetic local search algorithm:** Genetic algorithm is one of the random optimization algorithms which is suitable for complex problems in an unknown search space. It is also one of the evolutionary algorithms which was inspired by inheritance science. Its difference with other techniques lays in its initialization with random chromosome.

**Initial conditions:** Required initial information to start the GA method is as following:

- Population size: It is the number of chromosomes or scenarios which we will keep in each generation and we will give it N
- Number of replications: It is the number of simulated iterations of each scenario and we will give it n
- Crossover rate: This is the probability of performing crossover in the GA method, which is given  $P_c$
- Mutation rate: This is the probability of performing mutation in the GA method which is given  $P_M$

**Chromosome:** In GA method, chromosomes or scenarios are defined as a set of values for  $x_1, \dots, x_p$  which is a solution to the problem. Figure 3 shows chromosome.

**Initial population:** Generating an initial population of solutions is the first step to start the optimization process. The number of solutions is N and we select solutions

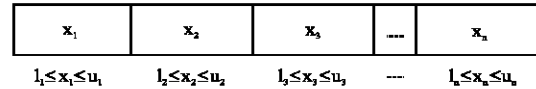


Fig. 3: Chromosome

randomly to cover a wide range of solution space. So we define the following parameters to simulate each gen:

- $x_{ij}$  : The input variable  $i$  in scenario  $j$ ,  $i = 1, \dots, p$   
 $j = 1, \dots, N$
- $L_x$  : Lower limit for input variable
- $U_x$  : Upper limit for input variable
- $\alpha$  : Random number between 0 and 1

$$X_{ij} = U_{X_i} - (U_{X_i} - L_{X_i}) \times \alpha$$

If  $\alpha$  has value 0, it means the related gen is in its upper bound and if it has value 1, the related gen is in its lower bound.

**Crossover, mutation and reproduction:** Genetic algorithm initiates with a population of chromosomes and also produces new chromosomes using crossover, mutation and reproduction operators. In a crossover process, it is necessary to mate pairs of chromosomes to create offspring. We implement it by selecting randomly a pair of chromosomes with probability of  $P_c$  from the population. In this study we apply two points crossover. In order to implement mutation operation we use pair wise exchange operator in which two genes are chosen randomly and exchanged. The reproduction operator chooses chromosomes with better values of fitness and puts them in a new population for the next generation.

**Evaluation:** After producing the new chromosomes by crossover and mutation operators, we evaluate each chromosome by suitable fitness function and use the result to reproduce a new generation.

**Chromosomes selection:** We have chosen roulette wheel approach from different methods of selection. In this method parents choosing depend on their fitness values thus better chromosomes have more opportunity for being selected.

**Parameters tuning:** There are several different factors which affect the algorithm in reaching to a suitable and desirable solution and parameters values is one of them. Whereas different parameters combinations would lead to different solutions, we try to find the most appropriate set of parameters in GA. This is called parameter tuning which

is considered to be the most important and perhaps the most difficult task. Reaching to a set of fixed values for each parameter of GA is a very challenging and difficult task while considering large size tests because of the multimodality and nonlinearity of different kind of objective functions.

To achieve this goal firstly we carry out an extensive experiment in order to determine effective parameters using design of experiments (DOE). Then an empirical study of various possible combinations of parameters will be done to define the best levels of those parameters.

**Solution algorithm:** Our suggested algorithm comprises two cycles: principal and internal. Outputs of the internal cycle, definition of lower and upper limit of each response, are inputs of principal cycle. One of the best features of this proposed algorithm is its long term memory. In each iteration the number of chromosomes with a low fitness value, would be saved in this memory which is used to fix a high diversity in the solution space. Another feature of this algorithm is that it uses shock operator. Since a small search space generally would lead to quick convergence into a local optima, we use shock operator to prevent it from getting stuck in a local optima. This operator highly influences the improvement of final solution. After some iteration, in case of having no improvement on the last solution, it will simulate some chromosomes and insert them in the population. In this way the search space would be extended because of generating diversity. Integration of genetic algorithm and local search is another feature of our proposed algorithm. This proposed algorithm has also a short term memory which is responsible for keeping the best solution. It would be

updated in two situations. First situation occurs when current iteration's solution is better than the solutions in this memory and the second situation happens when a fitness value in neighbors is better than the solutions of memory. Pseudo code of proposed algorithm is as it is shown in Fig. 4.

**NUMERICAL EXAMPLE**

Consider the following example with four design variables:

The ammoniac( $X_1$ ), the thickness of lead and expulsion alloys on the radiator pipe( $X_2$ ), temperature( $X_3$ ) and the percentage of expulsion in the alloy of the radiator pipe( $X_4$ ). The responses are corrosion ( $Y_1$ ) and adhesiveness ( $Y_2$ ). The design is a central composite design. Required data are shown in Table 2.

Correlation coefficient and the covariance matrix of the responses ( $Y_1 Y_2$ ) are as following:

$$R(Y_1, Y_2) = 0.994$$

$$S_2(Y_1, Y_2) = \begin{bmatrix} 1580.79 & 1582.55 \\ 1582.55 & 1599.04 \end{bmatrix}$$

Therefore the responses  $Y_1$  and  $Y_2$  are highly correlated. At first, we coded the above data and then we fitted a second-order model for both responses. The response surface curves for  $Y_1$  and  $Y_2$  is as following:

$$Y_1 = 250.34 - 34.26X_1 + 43.91X_2 - 1.55X_3 + 6.75X_4 - 22.97X_2^2 - 21.84X_3^2 - 23.22X_4^2 + 4.56X_1X_4 + 16.94X_2X_4 + 25.31X_3X_4$$

$$Y_2 = 176.75 - 2.01X_1 + 30.6X_2 + 2.59X_3 + 16.87X_4 - 13.31X_1^2 - 13.19X_2^2 - 12.94X_3^2 - 12.44X_4^2 + 8.56X_1X_4 - 3.94X_2X_4 - 8.69X_3X_4$$

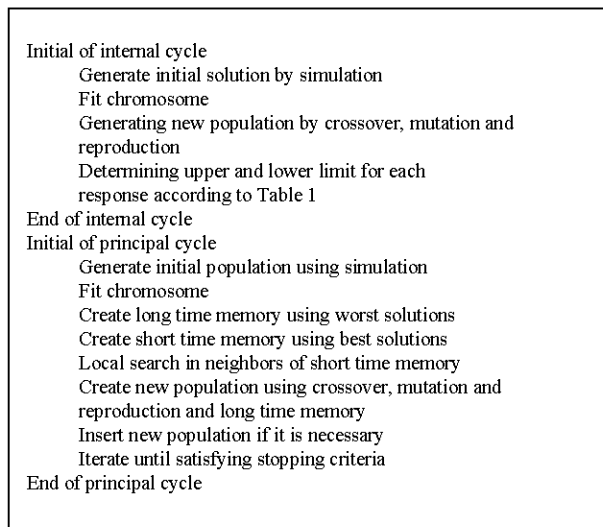


Fig. 4: Proposed algorithm's pseudo code

Table 2: Sampling data

Observation	$X_1$	$X_2$	$X_3$	$X_4$	$Y_1$	$Y_2$
1	2	18	330	25	52	50
2	7	18	360	25	50	45
3	7	23	330	25	120	117
4	2	23	360	25	170	159
4	7	18	330	30	120	110
6	2	18	360	30	94	90
7	2	23	330	30	186	178
8	7	23	360	30	180	176
9	4.5	20.5	345	27.5	166	160
10	4.5	20.5	345	27.5	165	163
11	4.5	20.5	345	27.5	167	165
12	4.5	20.5	345	27.5	161	166
13	4.5	20.5	345	31.04	172	169
14	4.5	20.5	345	23.96	160	157
15	4.5	24.04	345	27.5	173	174
16	4.5	16.96	345	27.5	155	150
17	4.5	20.5	366.2	27.5	171	167
18	4.5	20.5	323.8	27.5	157	159
19	8.035	20.5	345	27.5	169	161
20	0.965	20.5	345	27.5	162	159

**Table 3: Factorial levels**

Factors	Lower bound	Upper bound
Mutation rate	0.06	0.1
Crossover rate	0.6	0.9
Number of iteration	10	500
Population size	100	500

**Table 4: Optimal result of the existing methods and proposed method**

Methods	X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	X <sub>4</sub>	Y <sub>1</sub>	Y <sub>2</sub>
Derringer and Suich	0	1	0	0	271.3	194.16
Limited goals method	0	1	0.3802	0.7173	279.5	193.8
Pasandideh and Niaki	0.23	0.99	0.0004	0.49	284.6	195.74
Proposed algorithm	-0.5374	1	0.1816	1	291.3063	210.5819

The studied multi-response problem is as follows:

$$\begin{aligned} \text{Max: } Z_1 &= Y_1 \\ \text{Max: } Z_2 &= Y_2 \\ \text{s.t. } -1 &= X_i = 1 \quad i=1, 2, 3, 4 \end{aligned}$$

Four factors (Population size, Number of iteration, Crossover and Mutation rate) are considered as effective parameters with lower and upper limits which are shown in Table 3.

To tune parameters (effective factors) central composite design is used. Set of Results after optimization are presented below:

Population size: 300  
 Number of iteration: 225  
 Crossover rate: 75%  
 Mutation rate: 8%

We also implemented a comparison between the proposed and existing algorithms which is shown in Table 4.

As shown above, our proposed method has significant preference rather than other methods and presents a better solution.

### CONCLUSION

In this study, we modeled a multi-response statistical optimization problem through the fuzzy goal programming approach. Input data are generated through simulated system and we presented a methodology which is the combination of genetic and local search algorithm.

Our proposed algorithm is comprised of some distinctive characteristics such as having long and short term memory and shock operator. We further applied central composite design approach to tune parameters which is effective on algorithm's performance. Eventually results are compared with other recently reported methods, presented in literature and we showed our hybrid GLSA algorithm produced better performance. For future researches in this area, we recommend the followings:

- Applying other methods for generating input data like neural networks instead of simulation which may produce closer data to real life problems
- Using more factors for evaluating and using MCDM techniques to measure the performance of algorithm
- Using different crossover and mutation operators in the GA methods may lead to different conclusions
- Using other metaheuristic methods may be more efficient for this problem
- Performing other methods; for generating benchmark problem like Taguchi

### REFERENCES

Artiles-Leon, N., 1996. A pragmatic approach to multiple response problems using loss functions. *Qual. Eng.*, 9: 213-220.

Avello, A.E., F.F. Baesler and R.J. Moraga, 2004. A Metaheuristic based on simulated annealing for solving multiple objective problems in simulation optimization. *Proceeding the Winter Simulation Conference*, December 5-8, Winter Simulation Conference, pp: 508-513.

Chao, F.H., T.C. Jun and L.S. Yuan, 2007. A fuzzy goal programming approach to multi-objective optimization problem with priorities. *Eur. J. Oper. Res.*, 176: 1319-1333.

Chen, Y. and C. Liu, 1994. Multiobjective VAR planning using the goal attainment method. *Proc. Inst. Elect. Eng. Gen. Transm. Distrib.*, 141: 227-232.

Cheng, C.B., C.J. Cheng and E.S. Lee, 2002. Neuro-fuzzy and genetic algorithm in multiple response optimization. *Comput. Math. Appl.*, 44: 1503-1514.

Correia, D.S., C.V. Goncalves, S.S. Da Cunha and V.A. Ferraresi, 2005. Comparison between genetic algorithm and response surface methodology in GMAW welding optimization. *J. Mater. Proc. Technol.*, 160: 70-76.

Gen, M., K. Ida, Y. Tsujimura and C. Kim, 1993. Large-scale 0-1 fuzzy goal programming and its application to reliability optimization problem. *Comput. Ind. Eng.*, 24: 539-549.

Kapur, C.K. and B.R. Cho, 1996. Economic design of the specification region for multiple quality characteristics. *IIE Trans.*, 28: 237-248.

Khoo, L.P. and C.H. Chen, 2001. Integration of response surface methodology with genetic algorithms. *Int. J. Adv. Manuf. Technol.*, 18: 483-489.

Kim, D. and S. Rhee, 2004. Modelling and optimization of a GMA welding process by genetic algorithm and response surface methodology. *Int. J. Prod. Res.*, 40: 1699-1711.



- Koksoy, O. and T. Yalcinoz, 2006. Mean square error criteria to multi response process optimization by a new genetic algorithm. *Applied Math. Comput.*, 175: 1657-1674.
- Mirrazavi, K., D.F. Jones and M. Tamiz, 2001. A comparison of genetic and conventional methods for the solution of integer goal programmes. *Eur. J. Operat. Res.*, 132: 594-602.
- Montgomery, D.C., 1997. *Design and Analysis of Experiments*. 4th Edn. John Wiley and Sons, New York, ISBN 0-471-15746-5 pp: 487.
- Oktem, H., T. Erzurumlu and H. Kurtaran, 2005. Application of response surface methodology in the optimization of cutting conditions for surface roughness. *J. Mat. Processing Technol.*, 70: 11-16.
- Ozcelik, B. and T. Erzurumlu, 2005. Determination of effecting dimensional parameters on warpage of thin shell plastic parts using integrated response surface method and genetic algorithm. *Int. Adv. Manuf. Technol.*, 27: 468-472.
- Pasandideh, S.H.R. and S.T.A. Niaki, 2006. Multi-response simulation optimization using genetic algorithm within desirability function framework. *Applied Math. Comput.*, 175: 366-382.
- Pignatiello, J.J., 1993. Strategies for robust multi-response quality engineering. *IIE Trans.*, 25: 5-15.
- Ross, P.J., 1996. *Techniques for Quality Engineering*. 2nd Edn. McGraw-Hill, New York, ISBN-13: 978-0070539587.
- Suresh, P.V.S. P. Venkateswara Rao and S.G. Deshmukh, 2002. A genetic algorithm approach for optimization of surface roughness prediction model. *Int. J. Mach. Tools Manuf.*, 42: 675-680.
- Winston, W.L., 2003. *Operations Research: Applications and algorithms*. 4th Edn. Duxbury Press, ISBN-13: 9780534380588.