Volume 4, No. 8, August 2013
Journal of Global Research in Computer Science
REVIEW ARTICLE
Available Online at www.jgrcs.info
ISSN-2229-371X

GOAL GEOMETRIC PROGRAMMING PROBLEM (G^2P^2) WITH CRISP AND IMPRECISE TARGETS

Payel Ghosh*[1], Tapan Kumar Roy[2]
[*1] Department of Mathematics, Adamas Institute of Technology, P.O.-Jagannathpur, Barbaria, Barasat, North 24 Parganas, West Bengal - 700126, India, ghoshpayel86@yahoo.com
[2]Department of Mathematics, Bengal Engineering and Science University, Shibpur, P.O.-Botanic Garden, Howrah, West Bengal-711103, India, roy_t_k@yahoo.co.in

Abstract: There are very common and widely used forms of solving linear and non-linear Goal Programming Problem. They are Archimedean, Lexicographic and MINMAX etc. This paper proposes a Geometric Programming method to solve a non-linear Goal Programming Problem. In particular, it demonstrates a new approach goal geometric programming in both crisp and imprecise environment. There is a numerical example and also an application of this method in two-bar truss problem. Comparison with Kuhn-Tucker conditions and crisp goal geometric programming method in the numerical example, it shows the efficiency of this method. In this paper, we have described fuzzy goal geometric programming and also implemented it on the same numerical example like crisp goal geometric programming and two bar truss problem.

INTRODUCTION

Linear goal programming is widely used in decision making involving linear equations and multiple conflicting goals. In this model, the best solution is achieved when the sum of weighted deviations is minimal. But there are some real life situations like production planning, location distribution, risk management, chemical process design, pollution control and other engineering design where equations may be non-linear. For such type of mathematical model goal geometric programming may be an excellent method.

It is worthwhile to mention that, to solve goal-programming problems, the use of multi-objective optimization technique is not new (Romero 1991[1], Steuer 1986[2]). Deb (1999)[3] has developed an approach for solving non-linear goal programming problems by multi-objective genetic algorithms. Ojha et al. (2010[4][5]) has discussed geometric programming method to solve multi-objective programming problems using weighted sum method. Luptacik (2010)[6] described elaborately about single objective geometric programming and multi-objective geometric programming. Previously Romero (1991)[1], Tamiz et al. (1995)[7] and Tamiz and Jones (1997)[8] described weighted sum method in linear goal programming problem but in this paper we have used weighted sum method in goal programming problem with non-linear equations and solved it using geometric programming technique. Further Tamiz and Jones (2010)[9] have developed goal programming problem and applied it in health care and portfolio selection. Romero (2004)[10] has described general structure of achievement function in goal programming problem.

When goals are not precise, then fuzzy goals are introduced. Narasiman (1980)[11] first introduced fuzzy set theory in goal programming. Further the contribution of Tiwari et.al. (1984)[12], Kim, Whang (1998)[13], Ramik (1996)[14], Li (2012) [15], Ciptomulyono (2008) [16] are mentionable. Li, Hu (2009)[17] have used weighted sum method in fuzzy multiple objective goal programming problem and further developed it. Fuzzy goal programming has very extensive applications like portfolio selection, water quality management in river basin, structural optimization etc.

In this paper we have applied geometric programming technique to solve a nonlinear goal programming problem. There is a comparison of results between goal geometric programming technique and nonlinear programming technique applied on a numerical example. We have also applied this technique in "two-bar truss" problem (Rao 1996)[18]. Apart from the crisp goal geometric programming, fuzzy goal geometric programming is also discussed here. Fuzzy goal geometric programming is described here by the same numerical example and also applied on the same application as in crisp goal geometric programming.

Body Text:

Formulation of Multi-Objective programming:

A multi-objective non-linear programming can be written as

Find $x = (x_1, x_2 \ldots\ldots x_n)^T$ (1.1)

so as to Minimize: $f_{10}(x) = \sum_{i=1}^{P_{10}} c_{10i} \prod_{k=1}^{n} x_k^{a_{k10i}}$ with target c_{10} ,

Minimize: $f_{20}(x) = \sum_{i=1}^{P_{20}} c_{20i} \prod_{k=1}^{n} x_k^{a_{k20i}}$ with target c_{20} ,

……………………………………………

Minimize: $f_{m0}(x) = \sum_{i=1}^{P_{m0}} c_{m0i} \prod_{k=1}^{n} x_k^{a_{km0i}}$ with target c_{m0},

subject to $f_r(x) = \sum_{i=1}^{P_r} c_{ri} \prod_{k=1}^{n} x_k^{a_{kri}} \le c_r$, r = 1, 2, 3 … q

$x_k > 0$.k=1, 2, 3 … n

Where c_{j0i} are positive real numbers ∀ j=1, 2 …m; i=1, 2... P_r;

a_{kj0i} and a_{kri} are real numbers ∀ k=1, 2 …n; j=1, 2 …m; i=1, 2... P_r.

P_{j0}= Number of terms present in j0 th objective function.

P_r = Number of terms present in r th constraint.
c_r = Boundary value of the r th constraint.

In the above multi-objective programming model, there are m number of minimizing objective functions, q number of inequality type constraints and n number of strictly positive decision variables.

Formulation of Goal programming from Multi-Objective Programming:

Goal programming gives better results than ordinary multi-objective programming. To formulate the goal programming problem from the multi-objective programming problem, positive or negative deviations are minimized depending upon objective functions and constraints, e.g., for minimizing objective function positive deviation , for maximizing objective function negative deviation, for " $\le$ " type constraint positive deviation, for " $\ge$ " type constraint negative deviation is minimized.

From the multi-objective programming problem (1.1), goal formulation is given below:

Minimize $\sum_{j=1}^{m} d_{j0}^{+} + \sum_{r=1}^{q} d_r^{+}$ (2.1)
subject to f_{j0} (x) + d_{j0}^{-} - $d_{j0}^{+} = c_{j0}$, j = 1,2 …m
f_r (x) + d_r^{-} - d_r^{+} = c_r , r = 1, 2 …q
x_k > 0, k=1, 2, 3 …. n d_{j0}^{+} , d_{j0}^{-} , d_r^{+} , $d_r^{-} > 0$; $d_{j0}^{+} \times d_{j0}^{-}$= 0 ;
$d_r^{+} \times d_r^{-}$ = 0 .

d_{j0}^{+} = Positive deviation of minimizing objective function.
d_{j0}^{-} = Negative deviation of minimizing objective function.
d_r^{+} = Positive deviation of "$\le$" constraint.
d_r^{-} = Negative deviation of "$\le$" constraint.
c_{j0} , c_r are boundary values of objective function and constraint of (1.1).

The above model (2.1) can be transformed into
Minimize $\sum_{j=1}^{m} d_{j0}^{+} + \sum_{r=1}^{q} d_r^{+}$ (2.2)
subject to f_{j0} (x) - $d_{j0}^{+} \le c_{j0}$, j = 1,2 ….m
f_r (x) - $d_r^{+} \le c_r$, r = 1, 2 ……q
x_k >0, k=1, 2, 3 …. n
d_{j0}^{+} , $d_r^{+} > 0$.

The single solution like (x^*, d_{j0}^{+}, d_r^{+}), j =1,2…m; r =1,2,…q minimize the objective (2.2) and satisfy the constraints (2.3), (2.4), (2.5). But there are some cases where much more minimized value is required for any particular objectives or/and constraints. We usually tackle this situation by introducing weights. Give biggest weight (priority) for that deviation of the objective function or constraint for which we want to get more minimized value.

Then weighted goal formulation is as follows:
Minimize $\sum_{j=1}^{m} W_{j0}\, d_{j0}^{+} + \sum_{r=1}^{q} W_r d_r^{+}$, subject to
f_{j0} (x) - $d_{j0}^{+} \le c_{j0}$, j = 1, 2 …m
f_r (x) - $d_r^{+} \le c_r$, r = 1, 2 ……q
x_k > 0, k=1, 2, 3 …. n d_{j0}^{+} , $d_r^{+} > 0$,
and $\sum_{j=1}^{m} W_{j0} + \sum_{r=1}^{q} W_r = 1$; $W_{j0} > 0$,
$W_r > 0$

Goal programming gives pareto optimal solutions which is already described in Romero (1991) [1]. Now we prove a result concerning the pareto optimality of the solutions of weighted non-linear goal programming problem.

Theorem:

The solution of the following weighted goal programming problem
Minimize $\sum_{i=1}^{k} W_i\, d_i$,
subject to; $\sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} x_l^{a_{lir}}$ - $d_i \le \bar{C}_i$, for i= 1, 2 ... k
X= $\{x_l; l = 1, 2 \dots n\} \in S$; $d_i \ge 0$ for i= 1, 2 … k
is pareto optimal if d_i for each functions to be minimized have positive values at the optimum.

***Proof*:** Let $X^* \in S$ with positive deviational vector d^* (> 0) be the solution of the following weighted goal programming problem
Minimize $\sum_{i=1}^{k} W_i\, d_i$ (3.1)
subject to; $\sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} x_l^{a_{lir}}$ - $d_i \le \bar{C}_i$ for i= 1,2 ….. k
X= $\{x_l; l = 1, 2 \dots n\} \in S$; $d_i \ge 0$,for i= 1,2 … k
If possible let X^* is not Pareto optimal, so there exists a vector $X^0 \in S$ with positive deviational variable vector d^0 such that
$\sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^0)^{a_{lir}} \le \sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^*)^{a_{lir}}$ ∀ i=1, 2 ….. k (3.2)
$\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n} (x_l^0)^{a_{ljr}} < \sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n} (x_l^*)^{a_{ljr}}$ for at least one j (3.3)
From (3.3) $\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n} (x_l^*)^{a_{ljr}} - \sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n} (x_l^0)^{a_{ljr}}$ >0;
Let $\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n} (x_l^*)^{a_{ljr}} - \sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n} (x_l^0)^{a_{ljr}} = \beta$ (>0) (3.4)
We set $d_i^0 = d_i^*$ (>0) for i=1, 2 …… k (3.5)
and d_j^0= max (0, d_j^* - β) $\ge$ 0 and i $\ne$ j (3.6)

Here d_i^0 is the positive deviational variable corresponding to X^0 for i= 1,2 …. k
From (3.2), $\sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^0)^{a_{lir}} \le \sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^*)^{a_{lir}}$ (2.4)
Or, (2.5) $\sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^0)^{a_{lir}} - d_i^0 \le \sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^*)^{a_{lir}} - d_i^0$
$\le (\sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^*)^{a_{lir}} - d_i^*$
∀ i=1, 2 … k

[Using (3.5)]
[As X^* be the solution of (3.1), so $\sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^*)^{a_{lir}} - d_i^* \le \bar{C}_i$]
So $\sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^0)^{a_{lir}}$ - $d_i^0 \le (\sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^*)^{a_{lir}} - d_i^* \le \bar{C}_i$
i.e. $\sum_{r=1}^{p} c_{ir} \prod_{l=1}^{n} (x_l^0)^{a_{lir}}$ - $d_i^0 \le \bar{C}_i$ for i= 1, 2 … k but i $\ne$ j (3.7)
From (3.6), d_j^0 = max (0, d_j^* - β)
so $d_j^0 = d_j^* - \beta$ if $d_j^* - \beta > 0$ (3.8)
= 0 if $d_j^* - \beta \le 0$ (3.9)
Case 1: If $d_j^* - \beta > 0$, then
$\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n} (x_l^0)^{a_{ljr}} - d_j^0 = \sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n} (x_l^0)^{a_{ljr}} - d_j^* + \beta$ [by (3.8)]

$=\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^0)^{a_{ljr}} - d_j^* + (\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^*)^{a_{ljr}} - \sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^0)^{a_{ljr}}$

[by (3.4)]

$= (\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^*)^{a_{ljr}} - d_j^* \le \bar{C}_j$

[As X^* be the solution of (3.1), so $\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^*)^{a_{ljr}} - d_j^* \le \bar{C}_j$]

So $\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^0)^{a_{ljr}} - d_j^0 \le \bar{C}_j$ (3.10)

So X^0 satisfies the constraints of (3.1) [following (3.7) and (3.10)].

From (3.8) $d_j^0 = d_j^* - \beta < d_j^*$ [$\because \beta > 0$]

So using (3.5) $d_i^0 \le d_i^*$ $\forall$ i = 1, 2 …. K

Case 2: If $d_j^* - \beta \le 0$ then

$\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^0)^{a_{ljr}} - d_j^0 = \sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^0)^{a_{ljr}}$

$=\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^*)^{a_{ljr}} - \beta$ [by (3.4)]

$\le \sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^*)^{a_{ljr}} - d_j^*$ [by $d_j^* \le \beta$]

$\le \bar{C}_j$ (3.11)

[As X^* be the solution of (3.1), so $\sum_{r=1}^{p} c_{jr} \prod_{l=1}^{n}(x_l^*)^{a_{ljr}} - d_j^* \le \bar{C}_j$]

So X^0 satisfies the constraints of (3.1) [following (3.7) and (3.10)].

and $d_j^0 = 0 < d_j^*$ hence $d_j^0 < d_j^*$.

So using (3.5) $d_i^0 \le d_i^*$ $\forall$ i = 1,2 …. k

So for all positive weights W_i [for i= 1, 2 ……. k]

$\sum_{i=1}^{k} W_i\, d_i^0 < \sum_{i=1}^{k} W_i\, d_i^*$ (3.12)

So from (3.7), (3.10), (3.11) and (3.12), we have seen that X^0 is a solution of (3.1).

It contradicts the fact that X^* is a solution of (3.1). Hence X^* is Pareto optimal.

Formulation of Goal Geometric programming problem using weights:

A weighted goal-geometric programming problem can be written as:

Find x= $(x_1, x_2 \ldots\ldots x_n)^T$, d=(d_{j0}^+ , d_r^+)^T

subject to $\frac{f_{j0}(x)}{c_{j0}} - \frac{d_{j0}^+}{c_{j0}} \le 1$, j = 1,2 ….m

$\sum_{j=1}^{m} W_{j0} + \sum_{r=1}^{q} W_r = 1$; $W_{j0} > 0, W_r > 0$.

Dual form of Goal Geometric programming problem:

The model given by (4.1) is a normal geometric programming problem and it can be solved by using primal based algorithm for non-linear primal problem or dual programming.

The model given by (4.1) can be transformed to the corresponding dual geometric programming as:

Maximize d(δ)=

$$\xi_0[\prod_{j=1}^{m}\left(\frac{W_{j0}}{\delta_{j0}}\right)^{\delta_{j0}} \prod_{r=1}^{q}\left(\frac{W_r}{\delta_r}\right)^{\delta_r} \prod_{j=1}^{m}\prod_{i=1}^{P_{j0}}\left(\frac{C_{j0i}}{C_{j0}\,\delta_{ji}}\right)^{\delta_{ji}} \prod_{j=1}^{m}\left(\frac{1}{C_{j0}\delta_{j\,P_{j0}+1}}\right)^{-\delta_{j\,P_{j0}+1}} \prod_{j=1}^{m}\lambda_j(\delta)^{\lambda_j(\delta)} \prod_{r=1}^{q}\prod_{i=1}^{P_r}\left(\frac{C_{ri}}{C_r\delta_{ri}}\right)^{\delta_{ri}} \prod_{r=1}^{q}\left(\frac{1}{C_r\delta_{r\,P_r+1}}\right)^{-\delta_{r\,P_r+1}} \prod_{r=1}^{q}\lambda_r(\delta)^{\lambda_r(\delta)}]^{\xi_0}$$

subject to $\sum_{j=1}^{m}\delta_{j0} + \sum_{r=1}^{q}\delta_r = \xi_0$;

$\xi_0 = \pm 1$,

$\delta_{j0} - \delta_{j\,P_{j0}+1} = 0$, j =1, 2 … m,

$\delta_r - \delta_{r\,P_r+1} = 0$, r =1, 2 … q,

$\sum_{j=1}^{m}\sum_{i=1}^{P_{j0}} a_{kj0i}\delta_{ji} + \sum_{r=1}^{q}\sum_{i=1}^{P_r} a_{kri}\delta_{ri} = 0$; k= 1, 2 … n.

Where $\lambda_j(\delta) = \sum_{i=1}^{P_{j0}}\delta_{ji} - \delta_{j\,P_{j0}+1}$, j = 1, 2 … m,

$\lambda_r(\delta) = \sum_{i=1}^{P_r}\delta_{ri} - \delta_{r\,P_r+1}$, r = 1, 2 … q.

RESULTS AND DISCUSSIONS

Numerical Example1:

A multi-objective goal programming problem

Minimize $f_1(x_1, x_2) = x_1^{-1}\, x_2^{-2}$

with target value 4,

Minimize $f_2(x_1, x_2) = 2\, x_1^{-2}\, x_2^{-3}$

with target value 50,

subject to $x_1 + x_2 \le 1$,

$x_1 , x_2 \ge 0$.

The above multi-objective goal programming problem is converted into single objective goal geometric programming problem using deviations and giving the weights (priorities). The formulation is given below:

Minimize $W_1\, d_1^+ + W_2\, d_2^+$, (5.1)

subject to $\frac{x_1^{-1}x_2^{-2}}{4} - \frac{d_1^+}{4} \le 1$

$\frac{2\,x_1^{-2}\,x_2^{-3}}{50} - \frac{d_2^+}{50} \le 1$,

$x_1 + x_2 \le 1$,

$x_1, x_2, d_1^+, d_2^+ > 0$.

Illustration:

Degree of difficulty=8-(4+1) = 3.

Dual of (5.1) is given by:

Maximize **d** (δ) = (4.1)

$\frac{f_r(x)}{c_r} - \frac{d_r^+}{c_r} \le 1$, r = 1, 2 …. q x_k

$$\xi_0[\left(\frac{W_1}{\delta_{01}}\right)^{\delta_{01}} X \left(\frac{W_2}{\delta_{02}}\right)^{\delta_{02}} X \left(\frac{1}{4\delta_{11}}\right)^{\delta_{11}} X \left(\frac{1}{4\delta_{12}}\right)^{-\delta_{12}} X \left(\frac{2}{50\delta_{21}}\right)^{\delta_{21}} X \left(\frac{1}{50\delta_{22}}\right)^{-\delta_{22}} X \left(\frac{1}{\delta_{31}}\right)^{\delta_{31}} X \left(\frac{1}{\delta_{32}}\right)^{\delta_{32}} X\, \lambda_1(\delta)^{\lambda_1(\delta)} X\, \lambda_2(\delta)^{\lambda_2(\delta)} X\, \lambda_3(\delta)^{\lambda_3(\delta)}]^{\xi_0}$$

such that $\delta_{01} + \delta_{02} = \xi_0$ (5.1)

$\delta_{01} - \delta_{12} = 0$ (5.2)

$\delta_{02} - \delta_{22} = 0$ (5.3)

$\delta_{11} - 2\,\delta_{21} + \delta_{31} = 0$ (5.4)

$2\delta_{11} - 3\,\delta_{21} + \delta_{32} = 0$ (5.5)

$\lambda_1(\delta) = \delta_{11} - \delta_{12}$ (5.6)

$\lambda_2(\delta) = \delta_{21} - \delta_{22}$ (5.7)

$\lambda_3(\delta) = \delta_{31} + \delta_{32}$ (5.8)

If $\xi_0 = -1$, then from (5.1), $\delta_{02} = -1-\delta_{01}$. Since $\delta_{01} > 0$, therefore according to the relation δ_{02} is negative which contradicts the positivity condition of dual variables.

Hence let $\xi_0 = 1$, then from (5.1) - (5.8) we get the following equations:

δ_{02}= 1-δ_{01}; δ_{12} = δ_{01}; δ_{22} = 1-δ_{01}; δ_{31}= δ_{11} +2 δ_{21}; δ_{32}= 2δ_{11} +3 δ_{21}

$\lambda_1(\delta)$ = δ_{11} - δ_{01} ; $\lambda_2(\delta)$ = δ_{21} + δ_{01} -1 ; $\lambda_3(\delta)$ =3 δ_{11}+5δ_{21}

Therefore, Maximize **d(δ)=**

$$[\left(\frac{W_1}{\delta_{01}}\right)^{\delta_{01}} \text{x} \left(\frac{W_2}{1-\delta_{01}}\right)^{\delta_{01}} \text{x} \left(\frac{1}{4\delta_{11}}\right)^{\delta_{11}} \text{x} \left(\frac{1}{4\delta_{01}}\right)^{-\delta_{01}} \text{x} \left(\frac{2}{50\delta_{21}}\right)^{\delta_{21}}$$

$$\text{x} \left(\frac{1}{50(1-\delta_{01})}\right)^{-(1-\delta_{01})}$$

$$\text{x} \left(\frac{1}{\delta_{11}+2\,\delta_{21}}\right)^{\delta_{11}+2\,\delta_{21}} \text{x} \left(\frac{1}{2\delta_{11}+3\,\delta_{21}}\right)^{2\delta_{11}+3\,\delta_{21}} \text{x } (\boldsymbol{\delta_{11}} -$$

δ01)(δ11 − δ01) x ***(δ21+ δ01*** −1***)(δ21+ δ01*** −1***)*** x***(*3 *δ11*+5*δ21)(*3 *δ11*+5*δ21)*]**

$$=[\boldsymbol{w_1^{\delta_{01}}} \text{ x } \mathbf{4}^{\delta_{01}} \text{ x } \boldsymbol{w_2^{1-\delta_{01}}} \text{ x } \mathbf{50}^{(1-\delta_{01})} \text{x} \left(\frac{1}{4\delta_{11}}\right)^{\delta_{11}} \text{x}(\boldsymbol{\delta_{11}} -$$

δ01)(δ11 − δ01)x***250δ21δ21***x***(δ21+ δ01*** −1***)(δ21+ δ01*** −1***)***x ***1δ11+2 δ21δ11+2 δ21***x***12δ11 +3 δ212δ11 +3 δ21***

$$\text{x}(\mathbf{3}\ \boldsymbol{\delta_{11}} + \mathbf{5}\boldsymbol{\delta_{21}})^{(3\,\delta_{11}+5\delta_{21})}\ ] \qquad (5.9)$$

Taking log on both side of (5.9) and then partially differentiating with respect to $\boldsymbol{\delta_{01}}$, $\boldsymbol{\delta_{11}}$ and $\boldsymbol{\delta_{21}}$ respectively and using the conditions of finding optimal solution we get these sets of equations

log(4w_1)-log(50w_2)-log($\boldsymbol{\delta_{11}}$ - $\boldsymbol{\delta_{01}}$)+log($\boldsymbol{\delta_{21}}$+ $\boldsymbol{\delta_{01}}$ -1)= 0 (5.10)

-log(4$\boldsymbol{\delta_{11}}$)+log($\boldsymbol{\delta_{11}}$-$\boldsymbol{\delta_{01}}$)-log($\boldsymbol{\delta_{11}}$+2$\boldsymbol{\delta_{21}}$)-log(2$\boldsymbol{\delta_{11}}$+3$\boldsymbol{\delta_{21}}$)+log(3 $\boldsymbol{\delta_{11}}$ + 5$\boldsymbol{\delta_{21}}$)=0 (5.11)

log(2)-log(50$\boldsymbol{\delta_{21}}$)+ log($\boldsymbol{\delta_{21}}$+ $\boldsymbol{\delta_{01}}$ -1)-2log($\boldsymbol{\delta_{11}}$ +2 $\boldsymbol{\delta_{21}}$)-3log(2$\boldsymbol{\delta_{11}}$ +3 $\boldsymbol{\delta_{21}}$)+ 5log(3 $\boldsymbol{\delta_{11}}$ + 5$\boldsymbol{\delta_{21}}$)= 0 (5.12)

From primal dual relations:

$W_1\ d_1^+ = \delta_{01}$d(δ), (5.13)

$W_2\ d_2^+ = (1-\ \delta_{01})$ d(δ), (5.14)

$\boldsymbol{x_1} = \frac{\delta_{31}}{\delta_{31}+\ \delta_{32}}$, or, $\boldsymbol{x_1} = \frac{\delta_{11}+2\,\delta_{21}}{(3\delta_{11}+5\,\delta_{21})}$

$\boldsymbol{x_2} = \frac{\delta_{32}}{\delta_{31}+\ \delta_{32}}$, or, $\boldsymbol{x_2} = \frac{2\delta_{11}+3\,\delta_{21}}{(3\,\delta_{11}+\ 5\delta_{21})}$

$\frac{d_1^+}{4} = \frac{\delta_{12}}{\delta_{11}-\ \delta_{12}}$, or, $\boldsymbol{d_1^+} = \frac{4\delta_{01}}{(\delta_{11}-\ \delta_{01})}$

$\frac{d_2^+}{50} = \frac{1-\delta_{01}}{\delta_{21}+\ \delta_{01}-1}$, or, $\boldsymbol{d_2^+} = \frac{50(1-\delta_{01})}{(\delta_{21}+\ \delta_{01}-1)}$

Solving (5.10), (5.11) and (5.12) having different weights and putting into (5.15), (5.16), (5.17), (5.18) we get the list of values which are in table-1:

Table- 1: Optimal table of values of goal geometric programming problem (G^2P^2) of (5.1)

				Optimal values of objectives	
W_1	W_2	**Optimal Dual variables**	**Optimal Primal variables**	**1st objective** f1(x1, x2)	**2nd objective** f2(x1, x2)
0.1	0.9	δ_{01}^*= 0.03986910, δ_{02}^*=0.9601309 δ_{11}^*=0.09408702, δ_{12}^*=0.0398691 δ_{21}^*= 7.059638, δ_{22}^*=0.9601309 δ_{31}^*=14.21336,δ_{32}^* =21.36709	x_1^*= 0.3994711 x_2^*= 0.6005289 d_1^{+*}= 2.941396 d_2^{+*}=7.870561	6.941396	57.87054
0.2	0.8	δ_{01}^*= 0.08534347, δ_{02}^*=0.9146565 δ_{11}^*=0.2015470, δ_{12}^*=0.08534347 δ_{21}^*= 6.724859, δ_{22}^*=0.9146565 δ_{31}^*=13.65127,δ_{32}^* =20.57767	x_1^*= 0.3988224 x_2^*= 0.6011776 d_1^{+*}= 2.937724 d_2^{+*}=7.871124	6.937690	57.87121
0.3	0.7	δ_{01}^*= 0.1376842, δ_{02}^*=0.8623158 δ_{11}^*= 0.3254526, δ_{12}^*=0.1376842 δ_{21}^*= 6.338890, δ_{22}^*=0.8623158 δ_{31}^*=13.00323, δ_{32}^*=19.66758	x_1^*=0.3980077 x_2^*= 0.6019923 d_1^{+*}= 2.933064 d_2^{+*}=7.872767	6.933087	57.87277
0.4	0.6	δ_{01}^*= 0.1985779, δ_{02}^*=0.8014221 δ_{11}^*=0.4699326, δ_{12}^*=0.1985779 δ_{21}^*= 5.889316, δ_{22}^*=0.8014221 δ_{31}^*=12.24856,δ_{32}^* =18.60781	x_1^*=0.3969541 x_2^*=0.6030459 d_1^{+*}=2.927208 d_2^{+*}=7.875774	6.927220	57.87597
0.5	0.5	δ_{01}^*=0.2702759, δ_{02}^*=0.7297241 δ_{11}^*=0.6405820, δ_{12}^*=0.2702759 δ_{21}^*= 5.358551, δ_{22}^*=0.7297241 δ_{31}^*=11.35768,δ_{32}^* =17.35682	x_1^*=0.3955383 x_2^*= 0.6044617 d_1^{+*}= 2.919486 d_2^{+*}= 7.882387	6.919487	57.88240
0.6	0.4	δ_{01}^*=0.3559232, δ_{02}^*=0.6440768 δ_{11}^*=0.8453603, δ_{12}^*=0.355923 δ_{21}^*=4.722719, δ_{22}^*=0.6440768 δ_{31}^*=10.29080,δ_{32}^* =15.85888	x_1^*=0.3935344 x_2^*= 0.6064656 d_1^{+*}= 2.908837 d_2^{+*}= 7.895726	6.908837	57.89567
0.7	0.3	δ_{01}^*=0.4600038, δ_{02}^*=0.539996 δ_{11}^*=1.095969, δ_{12}^*=0.4600038 δ_{21}^*=3.946951, δ_{22}^*=0.5399962 δ_{31}^*=8.989871,δ_{32}^* =14.03279	x_1^*=0.3904792 x_2^*= 0.6095208 d_1^{+} *= 2.893264 d_2^{+*}= 7.924910	6.893266	57.92534
0.8	0.2	δ_{01}^*=0.5891060, δ_{02}^*=0.410894 δ_{11}^*=1.410602, δ_{12}^*=0.5891060 δ_{21}^*=2.978068, δ_{22}^*=0.4108940 δ_{31}^*=7.366738,δ_{32}^* =11.75541	x_1^*=0.3852464 x_2^*=0.6147536 d_1^{+*}=2.868455 d_2^{+*}=8.002847	6.868458	58.00288
0.9	0.1	δ_{01}^*=0.7542469, δ_{02}^*=0.245753 δ_{11}^*=1.822708, δ_{12}^*=0.7542469 δ_{21}^*=1.729726, δ_{22}^*=0.24575 δ_{31}^*=5.282160,δ_{32}^* =8.834594	x_1^*=0.3741767 x_2^*=0.6258233 d_1^{+*}=2.823676 d_2^{+*}=8.280242	6.823698	58.28024

Here, for different weights we get different optimum values of decision variables, deviation variables and objective functions. The logic of the theorem is also proved here since all the deviations are positive at the optimum. Hence the solutions are pareto optimal.

Solving (5.1) in non-linear programming (Kuhn-Tucker conditions) we get the following results:

Minimize $\boldsymbol{W_1}\ \boldsymbol{d_1^+} + \boldsymbol{W_2}\ \boldsymbol{d_2^+}$,

subject to $\frac{x_1^{-1}x_2^{-2}}{4} - \frac{d_1^+}{4} \le 1$,

$\frac{2\,x_1^{-2}\,x_2^{-3}}{50} - \frac{d_2^+}{50} \le 1$,

$\boldsymbol{x_1} + \boldsymbol{x_2} \le 1$,

$\boldsymbol{x_1}, \boldsymbol{x_2}, \boldsymbol{d_1^+}, \boldsymbol{d_2^+} \ge 0$.

Lagrangian L=($\boldsymbol{W_1}\ \boldsymbol{d_1^+} + \boldsymbol{W_2}\ \boldsymbol{d_2^+}$)+ λ_1 ($\frac{x_1^{-1}x_2^{-2}}{4} - \frac{d_1^+}{4}$ - 1)+λ_2($\frac{2\,x_1^{-2}\,x_2^{-3}}{50} - \frac{d_2^+}{50}$ - 1)+λ_3($\boldsymbol{x_1}+\boldsymbol{x_2}$-1)

(i) $\boldsymbol{W_1} - \lambda_1$=0, (ii) $\boldsymbol{W_2} - \lambda_2$=0,

(iii) -$\lambda_1 \mathbf{x_1^{-2}} \mathbf{x_2^{-2}}-4\lambda_2 \mathbf{x_1^{-3}}\ \mathbf{x_2^{-3}}$+$\lambda_3$=0,

(iv) -2$\lambda_1 \mathbf{x_1^{-1}} \mathbf{x_2^{-3}}-6\lambda_2 \mathbf{x_1^{-2}}\ \mathbf{x_2^{-4}}$+$\lambda_3$=0,

(v) λ_1($\frac{x_1^{-1}x_2^{-2}}{4} - \frac{d_1^+}{4}$ -1) = 0,

(vi) λ_2($\frac{2\,x_1^{-2}\,x_2^{-3}}{50} - \frac{d_2^+}{50}$ – 1)=0,

(vii) λ_3($\boldsymbol{x_1}+\boldsymbol{x_2}$-1) = 0,

(viii) $\frac{x_1^{-1}x_2^{-2}}{4} - \frac{d_1^+}{4}$ -1≤ 0,

(ix) $\frac{2\,x_1^{-2}\,x_2^{-3}}{50} - \frac{d_2^+}{50}$ - 1≤0,

(x) $\boldsymbol{x_1}+\boldsymbol{x_2}$-1≤0, (xi) $\lambda_1, \lambda_2, \lambda_3 \ge 0$.

From (i) and (ii) λ_1=$\boldsymbol{W_1}$, λ_2=$\boldsymbol{W_2}$; $\boldsymbol{W_1}, \boldsymbol{W_2} \neq 0$.

Case 1: Let λ_3=0, then set of solutions comes from (iii), (iv), (v), (vi) is infeasible.

Case 2: Let $\lambda_3 \neq 0$, then solving (v), (vi), and (vii) we get $\boldsymbol{x_1}$ = 0.8230774, $\boldsymbol{x_2}$ = 0.1769226, $\boldsymbol{d_1^+}$ = 34.81438, $\boldsymbol{d_2^+}$ = 483.0876,

1st Objective $f_1(x_1, x_2)$= 38.81438, 2nd Objective $f_2(x_1, x_2)$= 533.0877.

Here is the comparison of results between goal geometric programming method and non-linear programming (Kuhn-Tucker conditions) which shows that goal geometric programming gives better result than non-linear programming. We compare the values of two objective functions of two different approaches. We take the values of primal variables, 1st and 2nd objective functions for equal weights from the above table and the values which we get in non-linear programming approach.

Table-2: Comparison of G²PM (Goal Geometric Programming Method) and NLPM (Non-linear programming method).

Approaches	**Primal variables**	**Dual Variables**	**1st objective** $f_1(x_1, x_2)$	**2nd objective** $f_2(x_1, x_2)$
G²PM(Goal Geometric Programming Method)	$\boldsymbol{x_1}$*=0.3955383 $\boldsymbol{x_2}$*= 0.604462 $\boldsymbol{d_1^+}$*= 2.919486 $\boldsymbol{d_2^+}$*= 7.882387	$\boldsymbol{\delta_{01}}^*$=0.2702759, $\boldsymbol{\delta_{02}}^*$=0.7297241 $\boldsymbol{\delta_{11}}^*$=0.6405820, $\boldsymbol{\delta_{12}}^*$=0.2702759 $\boldsymbol{\delta_{21}}^*$= 5.358551, $\boldsymbol{\delta_{22}}^*$=0.7297241 $\boldsymbol{\delta_{31}}^*$=11.35768,$\boldsymbol{\delta_{32}}^*$=17.35682	6.919487	57.88240
NLPM(Non-linear Programming Method)	$\boldsymbol{x_1}$= 0.8230774 $\boldsymbol{x_2}$= 0.1769226 $\boldsymbol{d_1^+}$= 34.81438 $\boldsymbol{d_2^+}$ = 483.0876	________________	38.81438	533.0877

Application on "Two bar truss problem":

The two bar truss is subjected to a vertical load 2P and is to be designed for minimum weight. The members have a tubular section with mean diameter d and wall thickness t and the maximum permissible stress in each member (σ_0) is equal to 60,000 psi. There are two goals:

Goal 1: weight should be near to 3

Goal 2: Ratio between stress and maximum permissible stress should be around 1.

Formulate the above goal programming problem and determine the values of mean diameter d and height h for the following data: P=33,000 lb, t=0.1 in. b=30 in. σ_0=60,000 psi, density ρ = 0.3 lb/in³.

Illustration: Weight =2$\rho\pi$ d t $\sqrt{b^2+h^2}$ =0.188d $\sqrt{900+h^2}$

Stress σ = (P$\sqrt{b^2+h^2}$) /(πd t h) =(33,000 $\sqrt{900+h^2}$)/ (πd h ×0.1)

Let $\sqrt{900^2+h^2}$ =y, or y^2=900 + h^2 .

Hence the new constraint is$(900+h^2)/y^2$ ≤1.

Therefore according to the first goal, weight 0.188 yd should be less than 3.

And the second goal is $\frac{\sigma}{\sigma_0}$ = (33,000 y) / (πd h ×0.1× 60,000) should be less than 1.

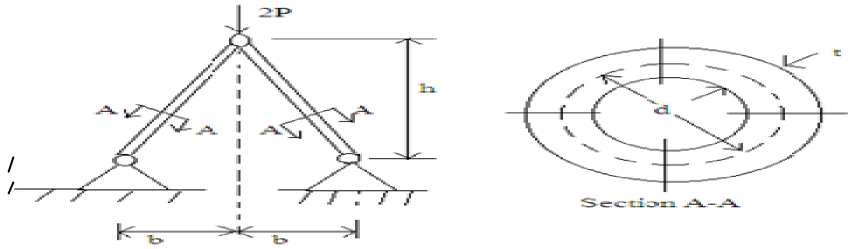

Fig-1:Two-bar truss under load

Therefore the multi objective nonlinear programming problem is

Minimize f_1=0.188 yd with target value 3 (6.1)

Minimize f_2=1.75 y $d^{-1}h^{-1}$ with target value 1
subject to 900 $y^{-2} + h^2y^{-2} \le 1$.
y, d, h>0.

The above model (6.1) can be converted into goal programming model as
Minimize $W_1 d_1^+ + W_2 d_2^+$ (6.2)
Subject to, 0.188 yd - $d_1^+ \le 3$
1.75 y $d^{-1}h^{-1}$- $d_2^+ \le 1$
900 $y^{-2} + h^2y^{-2} \le 1$.
y, d, h, d_1^+, d_2^+, W_1, W_2>0. W_1+W_2=1.

We solve the above model (6.2) converting it into geometric programming problem and then solving it we get the following results:

Table: 3 List of values of decision variables of two bar truss problem

				Optimal values	
W_1	W_2	Optimal Dual variables	Optimal Primal variables	1st objective (f_1)	2nd objective (f_2)
0.5	0.5	δ_{01}^*=0.295303, δ_{02}^*=0.7046970, δ_{11}^*=0.909394, δ_{12}^*=0.295303, δ_{21}^*=0.909394, δ_{22}^*=0.704697, δ_{31}^*=0.454697, δ_{32}^* =0.454697.	y^*=42.42641, d^*=0.5569888, h^*=30, d_1^{+*}=1.442634, d_2^{+*}=3.442634.	4.442634	4.44331

From the table, it is clear that goals of decision maker (DM) satisfy here appropriately. The first goal i.e. the first objective should be near to 3 and the second goal i.e. the second objective should be around 1, is maintained here. The logic of the theorem is also proved here since all the deviations are positive at the optimum. Hence the solutions are Pareto optimal.

Body of Text:

Fuzzy Goal Programming formulations:

In fuzzy environment the multi-objective goal programming problem can be written as
Find x = $(\boldsymbol{x_1}, \boldsymbol{x_2} \dots \boldsymbol{x_n})^T$ (7.1)
so as to $\widetilde{Minimize}$: $\boldsymbol{f_{10}}$ (x) = $\sum_{i=1}^{P_{10}} \boldsymbol{c_{10i}} \prod_{k=1}^{n} \boldsymbol{x_k}^{a_{k10i}}$ with target $\boldsymbol{c_{10}}$ and tolerance t_{10},
$\widetilde{Minimize}$: $\boldsymbol{f_{20}}$ (x) = $\sum_{i=1}^{P_{20}} \boldsymbol{c_{20i}} \prod_{k=1}^{n} \boldsymbol{x_k}^{a_{k20i}}$ with target $\boldsymbol{c_{20}}$ and tolerance t_{20},
……………………………………
$\widetilde{Minimize}$: $\boldsymbol{f_{m0}}$ (x) = $\sum_{i=1}^{P_{m0}} \boldsymbol{c_{m0i}} \prod_{k=1}^{n} \boldsymbol{x_k}^{a_{km0i}}$ with target $\boldsymbol{c_{m0}}$ and tolerance t_{m0},
Subject to f_r (x) = $\sum_{i=1}^{P_r} \boldsymbol{c_{ri}} \prod_{k=1}^{n} \boldsymbol{x_k}^{a_{kri}} \le \boldsymbol{c_r}$, r = 1, 2, 3 ... q
$\boldsymbol{x_k} > 0$. k =1, 2, 3, … n

There are various kinds of membership functions such that linear, exponential, hyperbolic piecewise linear etc. The corresponding membership functions of the minimize and maximize objectives of (7.1) are
$\mu(f_{j0}(X))$ = 1, if $f_{j0}(X) \le C_{j0}$
= $\frac{C_{j0}+t_{j0}-f_{j0}(X)}{t_{j0}}$, if $C_{j0} \le f_{j0}(X) \le C_{j0} + t_{j0}$
= 0, if $f_{j0}(X) \ge C_{j0} + t_{j0}$;
for j=1, 2 …m.

Now a crisp mathematical programming is made by substituting the membership functions. A weighted sum of membership function of multi-objective fuzzy goals is taken as achievement function in this paper. There are many practical situations where DM has different requirement for each objective function in multiple objective optimization problems according to his or her preference. Then there is a big role of weight factors. Hence the crisp programming model with weighted additive objective function is as follows:
Maximize v (μ)= $\sum_{j=1}^{m} W_j \ \mu(f_{j0}(X))$
Subject to $f_r(X) \le C_r$, r=1, 2 …q
$\mu(f_{j0}(X)) \le 1$
$\boldsymbol{x_k} > 0$, k=1, 2, 3 … n, $\sum_{j=1}^{m} W_j$ =1.

RESULTS AND DISCUSSIONS

Numerical Example 2:

A multi-objective fuzzy goal programming
$\widetilde{Minimize}$ $f_1(x_1, x_2)$ = $\boldsymbol{x_1^{-1}x_2^{-2}}$ with target value 4 tolerance 0.02,$\widetilde{Minimize}$ $f_2(x_1, x_2)$ = $\boldsymbol{2\ x_1^{-2}x_2^{-3}}$ with target value 50 tolerance 0.05,
subject to $\boldsymbol{x_1} + \boldsymbol{x_2} \le \boldsymbol{1}$, $\boldsymbol{x_1}$, $\boldsymbol{x_2} \ge 0$.

Membership function
$\mu_{g_1}(x)$ = 1, $\boldsymbol{x_1^{-1}x_2^{-2}} \le 4$
= $\frac{4-x_1^{-1}x_2^{-2}}{0.02}$ 4≤ $\boldsymbol{x_1^{-1}x_2^{-2}} \le 4.02$
= 0 $\boldsymbol{x_1^{-1}x_2^{-2}} \ge 4.02$
$\mu_{g_2}(x)$ = 1, $2\boldsymbol{x_1^{-2}x_2^{-3}} \le 50$
= $\frac{50-2x_1^{-2}x_2^{-3}}{0.05}$ 50≤ $\boldsymbol{2x_1^{-2}x_2^{-3}} \le 50.05$
= 0 $\boldsymbol{2x_1^{-2}x_2^{-3}} \ge 50.05$
Using fuzzy additive method:
Max $W_1(\frac{4-x_1^{-1}x_2^{-2}}{0.02})$ +$W_2(\frac{50-2x_1^{-2}x_2^{-3}}{0.05})$
subject to $\boldsymbol{x_1} + \boldsymbol{x_2} \le 1$; $\boldsymbol{x_1}$, $\boldsymbol{x_2} \ge 0$.
i.e. Min $W_1 \frac{x_1^{-1}x_2^{-2}}{0.02}$ +$W_2 \frac{2x_1^{-2}x_2^{-3}}{0.05}$ -200W_1- 1000W_2
(8.1)
subject to $\boldsymbol{x_1} + \boldsymbol{x_2} \le 1$; $\boldsymbol{x_1}$, $\boldsymbol{x_2} \ge 0$.

Here is the solution of the above model using geometric programming technique.

Table-4: Optimal values of decision variables of (8.1)

				Optimal values of objectives	
W_1	W_2	**Optimal Dual variables**	**Optimal Primal variables**	**1st objective** $f_1(x_1, x_2)$	**2nd objective** $f_2(x_1, x_2)$
0.1	0.9	δ_{01}^*=0.03222387, δ_{02}^*=0.9677761, δ_{11}^*=1.967776, δ_{12}^*=2.967776.	x_1^*=0.3986942, x_2^*=0.6013058.	6.936962	57.87140
0.2	0.8	δ_{01}^*=0.06961029, δ_{02}^*=0.9303897, δ_{11}^*=1.930390, δ_{12}^*=2.93039.	x_1^*=0.3971358, x_2^*=0.6028642.	6.928225	57.87532
0.3	0.7	δ_{01}^*=0.1135142, δ_{02}^*=0.8864858, δ_{11}^*=1.886486, δ_{12}^*=2.886486.	x_1^*=0.3952435, x_2^*=0.6047565.	6.917898	57.88405
0.4	0.6	δ_{01}^*=0.1658144, δ_{02}^*=0.8341856, δ_{11}^*=1.834186, δ_{12}^*=2.834186.	x_1^*=0.3928963, x_2^*=0.6071037.	6.905519	57.90092
0.5	0.5	δ_{01}^*=0.2291976, δ_{02}^*=0.7708024, δ_{11}^*=1.770802, δ_{12}^*=2.770802.	x_1^*=0.3899068, x_2^*=0.6100932.	6.890438	57.93217
0.6	0.4	δ_{01}^*=0.3076558, δ_{02}^*=0.6923442, δ_{11}^*=1.692344, δ_{12}^*=2.692344.	x_1^*=0.3859668, x_2^*=0.6140332.	6.871734	57.99019
0.7	0.3	δ_{01}^*= 0.4074204, δ_{02}^*=0.5925796, δ_{11}^*=1.592580, δ_{12}^*=2.592580.	x_1^*=0.3805302, x_2^*=0.6194698.	6.848108	58.10203
0.8	0.2	δ_{01}^*=0.5389038, δ_{02}^*=0.4610962, δ_{11}^*=1.461096, δ_{12}^*=2.461096.	x_1^*=0.3725203, x_2^*=0.6274797.	6.817901	58.33525
0.9	0.1	δ_{01}^*=0.7214652, δ_{02}^*=0.2785348, δ_{11}^*=1.278535, δ_{12}^*=2.278535.	x_1^*=0.3594349, x_2^*=0.6405651.	6.780367	58.89788

Application on "Two-bar truss problem":

The two bar truss is subjected to a vertical load 2P and is to be designed for minimum weight. The members have a tubular section with mean diameter d and wall thickness t and the maximum permissible stress in each member (σ_0) is equal to 60,000 psi. There are two goals:

Goal 1: Weight should be minimized with target value 3. Decision maker gives some relaxation of target value i.e.1 and sets his opinion that this goal is 'very important'.

Goal 2: Ratio between stress and maximum permissible stress should be minimized with target value 1. Here decision maker's opinion that the goal is also 'very important' and gives a relaxation 0.5 on the target value.

Formulate the above goal programming problem and determine the values of mean diameter d and height h for the following data: P=33,000 lb, t=0.1 in. b=30 in. σ_0=60,000 psi, density ρ = 0.3 lb/in^3.

Illustration:

Weight =$2\rho\pi$ d t $\sqrt{b^2+h^2}$ =0.188d $\sqrt{900+h^2}$

Stress σ = (P$\sqrt{b^2+h^2}$) /(πd t h) =(33,000 $\sqrt{900+h^2}$)/ (πd h $\times$0.1)

Let $\sqrt{900^2+h^2}$ =y, or y^2=$900+h^2$.

Hence the new constraint is$(900+h^2)/y^2 \le$1.

Therefore according to the first goal, weight 0.188 yd should be minimized with target value 3 and tolerance 1.

And the second goal is $\frac{\sigma}{\sigma_0}$ = (33,000 y) / (πd h $\times$0.1$\times$ 60,000) should be minimized with target value 1 and tolerance 0.5.

The fuzzy goal programming formulation is

$\widetilde{Minimize}$ 0.188 yd with target 3 tolerance1

$\widetilde{Minimize}$ 1.75 y $d^{-1}h^{-1}$ with target 1 tolerance 0. 5

900 $y^{-2}+h^2y^{-2} \le$ 1, y, d, h>0.

Membership function

μ_{g_1} = 1, 0.188 yd ≤ 3

$= \frac{3-0.188\text{ yd}}{1}$ $3\le$ 0.188 yd ≤ 4

= 0 0.188 yd ≥ 4

μ_{g_2}= 1, 1.75 y $d^{-1}h^{-1} \le \mathbf{1}$

= $\frac{1-1.75\text{ y }d^{-1}h^{-1}}{0.5}$ $1\le$ 1.75 y $d^{-1}h^{-1} \le 1.5$

= 0 1.75 y $d^{-1}h^{-1} \ge 1.5$

Using fuzzy additive method:

Maximize $W_1(3-0.188$ yd) + W_2(2 - 3.5 y $d^{-1}h^{-1}$)

subject to, $3-0.188$ yd $\le$1

2 - 3.5 y $d^{-1}h^{-1} \le$1

900 $y^{-2}+h^2y^{-2} \le$ 1, y, d, h>0.

That is the above model can be written as Minimize $W_1$0.188 yd + $W_2$3.5 y $d^{-1}h^{-1}$-3W_1-2W_2

subject to 10.63829$y^{-1}d^{-1} \le$ 1

0.28571y^{-1}h d $\le$ 1

900 $y^{-2}+h^2y^{-2} \le$ 1, y, d, h>0.

We solve the above crisp programming arranging it in to geometric programming problem ignoring (-3W_1-2W_2) and get the solution

Table: 5 List of values of decision variables of two bar truss problem

				Optimal values	
W_1	W_2	**Optimal Dual variables**	**Optimal Primal variables**	**1st objective** (f_1)	**2nd objective** (f_2)
0.5	0.5	δ_{01}^*=0.6666698, δ_{02}^*=0.3333302, δ_{11}^*=0.5116211, δ_{21}^*=0.1782818, δ_{31}^*=0.07752437, δ_{32}^* =0.07752437.	y^*=3.168602, d^*=3.357408, h^*=3.303224,	1.999999	0.4999925

Observing the results of crisp goal geometric programming on truss bar problem and fuzzy goal geometric programming on the same we can conclude that fuzzy goal geometric programming gives better result than crisp goal geometric programming. The first goal i.e. the weight should be minimized with target 3 which is 'very important' according to decision maker's choice and this target fulfills properly. Also the second goal i.e. the ratio of stress and maximum permissible stress should be minimized with target 1 which is also 'very important' according to decision maker's choice and this target also fulfills properly.

CONCLUSIONS

By using Goal Geometric Programming we can solve multi-objective goal programming problem (MOGPP). This method is very useful for many real life situations where equations are non-linear. We show the efficiency of this method with weighted sum deviations where weights (priorities) can be changed as per requirement. The variation of result according to weights (priorities) also shows the perfection of this method. Comparing with non-linear Optimization (Kuhn-Tucker conditions), this method gives better result which is already described in this paper. This method has several types of applications in the field of engineering, sciences etc. Here we have applied this method in two bar truss problem and show that this method is more efficient than the previously solved process. We have also discussed this method goal geometric programming in fuzzy environment and applied this on two bar truss problem. It is clear to us that fuzzy goal geometric programming gives better result i.e. this method satisfies all our targets and requirement as per our opinion. Instead of weighted sum method one can use weighted product method, MINMAX method etc. We can apply this method in imprecise environment like intuitionistic fuzzy and interval number also.

ACKNOWLEDGMENTS

It's my privilege to thank my respected guide for his enormous support and encourage for the preparation of this research paper. The great support of my family members makes me possible to do it.

REFERENCES

[1]. C. Romero, "Handbook of critical issues in goal programming", Oxford: Pergamon Press, 1991.

[2]. R. E. Steuer, "Multiple criteria optimization: Theory. Computation and application", New York: Wiley, 1986.

[3]. K. Deb, "Solving Goal programming problems using Multi-objective Genetic Algorithms", IEEE, 1999.

[4]. A.K. Ojha and K. K. Biswal, "Multi-Objective Geometric programming problem with weighted mean method", International Journal of Computer Science and Information Security Vol. 7 (No. 2), 2010.

[5]. A. K. Ojha and A. K. Das, "Multi-Objective Geometric programming problem being cost coefficients as continuous function with weighted mean method", Journal of Computing Vol. 2, (Issue 2) ISSN 2151-9617, 2010.

[6]. M. Luptacik, "Mathematical Optimization and Economic Analysis", Springer, New York: USA, 2010.

[7]. M. Tamiz, D. F. Jones and E. El-Darzi, "A review of goal programming and its applications", Annals of Operations Research 58: 39-53, 1995.

[8]. M. Tamiz and D. F. Jones, "Interactive frameworks for investigation of goal programming models: theory and practice" Journal of multi-criteria decision analysis 6: 52-60.

[9]. D. Jones and M. Tamiz, "Practical Goal Programming", International Series in Operations Research & Management Science, Springer, 2010.

[10]. C. Romero, "A general structure of achievement function for a goal programming model", European Journal of Operational Research 153: 675–686, 2004.

[11]. R. Narasimhan, "Goal Programming in Fuzzy Environment", Decision Sciences, 11: 325-336, 1980.

[12]. R. N. Tiwari, S. Dharmar S and J. R. Rao, " Priority structure in Fuzzy Goal Programming", Fuzzy sets and systems, 19:251-259, 1986.

[13]. J. S. Kim and K. S. Whang, "The Tolerance Approach to the Fuzzy Goal Programming Problems with Unbalanced Triangular Membership Function", European Journal of Operational Research, 107: 614-624, 1998.

[14]. J. Ramik, "New Interpretation of the inequality relations in fuzzy goal programming problems", Central European J. Oper. Res. Econom., 4(4):252-265, 1996.

[15]. G. Li, "Fuzzy goal programming – A parametric approach", Information Sciences, 195: 287–295, 2012.

[16]. U. Ciptomulyono, "Fuzzy Goal Programming Approach for Deriving Priority Weights in the Analytical Hierarchy Process (AHP) Method", Journal of Applied Sciences Research, 4(2): 171-177, 2008.

[17]. S. Li and C. Hu, "Satisfying optimization method based on goal programming for fuzzy multiple objective optimization problem", European Journal of Operational Research, 197: 675–684, 2009.

[18]. S. S. Rao, "Engineering optimization Theory and Practice", New Age International (P) Limited, Publisher, New Delhi: India, 1996.

[19]. K. Miettinen, "Nonlinear Multi-objective Optimization", Kluwer Academic Publishers, Boston, 1999.

[20]. R. K. Verma, "Fuzzy Geometric Programming with several objective functions", Fuzzy Sets and Systems, Vol. 35 (Issue 1): 115-120, 1990.

Short Bio-Data of All Authors

Payel Ghosh, assistant professor of Adamas Institute of Technology, Barasat, West Bengal, India, has completed M.Sc. from Bengal Engineering and Science University, Shibpur at 2009. She has completed her bachelor degree in Mathematics at 2007.

Tapan Kumar Roy, professor of Bengal Engineering and Science University, Shibpur, has published lots of papers on Fuzzy and Intuitionistic Fuzzy set Theory, Inventory, Transportation, Reliability Optimization, Portfolio Optimization, Fuzzy and Stochastic Optimization, etc.