Iraqi Journal of Statistical Science (20) 2011
The Fourth Scientific Conference of the College of Computer Science & Mathematics
pp [35-51]

Multi-Objective GPP with General Negative Degree of Difficulty: New Insights

Dr. Abbas Y. Al-Bayati* and Dr. Huda E. Khalid**

Abstract

The methods for solving nonlinear multi-objective optimization are divided into three major categories: methods with apriori articulation of preferences, methods with a posteriori articulation of preferences, and methods with no articulation of preferences. Really there is no single approach is superior. In this paper, a combination between two well known approaches has been used to solve multi-objective GP problems having negative degree of difficulty. First, we use an alternative procedure for converting GP problem having negative degree of difficulty to positive degree of difficulty; second we proposed to discuss all available cases for any number of multi-objective in GP problems using Lexicographic method. This avoids the difficulty of non-differentiability of the dual objective function in the classical methods.

البرمجة الهندسية المتعددة الأهداف مع درجات الصعوبة السالبة المعممة: رؤى جديدة

المستخلص

إن طرائق حل مسائل الامثلية غير الخطية والمتعددة الأهداف تنقسم إلى ثلاثة أنواع رئيسية: طرائق تتخذ تفضيلات مسبقة، طرائق تتخذ أفضليات لاحقة، وطرائق خالية من الافضليات. والحقيقة لا توجد أية طريقة من الطرائق المذكورة انفا تعتبر هي الافضل وخالية من العيوب. في بحثنا هذا دمجنا بين طريقتين معروفتين لاستخدامهما في حل مسائل البرمجة الهندسية بدوال هدف متعددة ذات درجة صعوبة

* Prof.\ College of Computers Sciences and Mathematics, Mosul University

** Lecturer\

Received:1/10 /2011 ____________________Accepted: 21 /12 / 2011

سالبة (d). أولا سنستخدم طريقة تكرارية لتحويل المسالة من درجة صعوبة سالبة الى درجة صعوبة موجبة، ثانيا سنقترح مناقشة كل الحالات المتوافرة لهذا النوع من المسائل باستخدام طريقة ال Lexicographic وبذلك نكون قد تجنبنا دوال الهدف غير القابلة للاشتقاق في مسائل الـ dual.

Keyword:- Geometric programming, multi-objective optimization, priori method, lexicographic ordering.

MSC Classification: **49M37, 90C30, 90C29**

1- Introduction:

1.1: **Geometric Programming**: The primal posynomial geometric programming problem (P) was originally described by (Duffin, et al., 1967).

$$(P) \qquad \min f_0(x) = \sum_{j=1}^{m_0} c_j \prod_{i=1}^{n} x_i^{a_{ji}} \qquad (1)$$

$$s.t. \; f_k(x) = \sum_{j=m_{k-1}+1}^{m_k} c_j \prod_{i=1}^{n} x_i^{a_{ji}} \leq 1 \quad k = 1, \dots, p \qquad (2)$$

$$x = (x_1, \dots, x_n) > 0 \qquad (3)$$

where:

$c_j > 0, \quad m_0 = no. of\ termes\ in\ the\ objective\ function,$

$j = 1, \dots, m;\ m_p = m\ ;\ m_k > m_{k-1}\ \forall k; a_{ji} \in R$

$i = 1, \dots, n\ ; j = 1, \dots, m.$

there are $p + 1$ posynomial expressions, one for the objective function (1) and p for inequality constraints (2). The number of posynomials in each constraints is $m_k - m_{k-1}$ for each $k = 1, \dots, p$, but m_0 for the objective function. The total number of terms in the primal program (P) is $m_p = m$.

1.2: Degree of Difficulty: the degree of difficulty of a geometric programming problem is the number of dual variables minus the number of dual equality constraints. If zero (and assuming the system of liner equality constraints of the dual has full rank), then there is a unique dual feasible solution. If the degree of difficulty positive, then the dual feasible region must be searched to maximize the dual objective, while if the degree of difficulty is negative, the dual constraints may be inconsistent.

2. The Foundation of Fundamental Concepts:

2.1 Multi-Objective Optimization:

Many decision and planning problems involve multiple conflicting objectives that should be considered simultaneously (alternatively, we can talk about multiple conflicting criteria). Such problems are generally known as multiple criteria decision making (MCDM) problems. We can classify MCDM problems in many ways depending on the characteristics of the problem in question. For example, we talk about multi-attribute decision analysis if we have a discrete, predefined set of alternatives to be considered. Here we study multi-objective optimization (also known as multi-objective mathematical programming) where the set of feasible solutions is not explicitly known in advance but it is restricted by constraint functions. In multi-objective optimization problems, it is characteristic that no unique solution exists but a set of mathematically equally good solutions can be identified. These solutions are known as non dominated, efficient, non inferior or Pareto optimal solutions, (Branke et al., 2008).

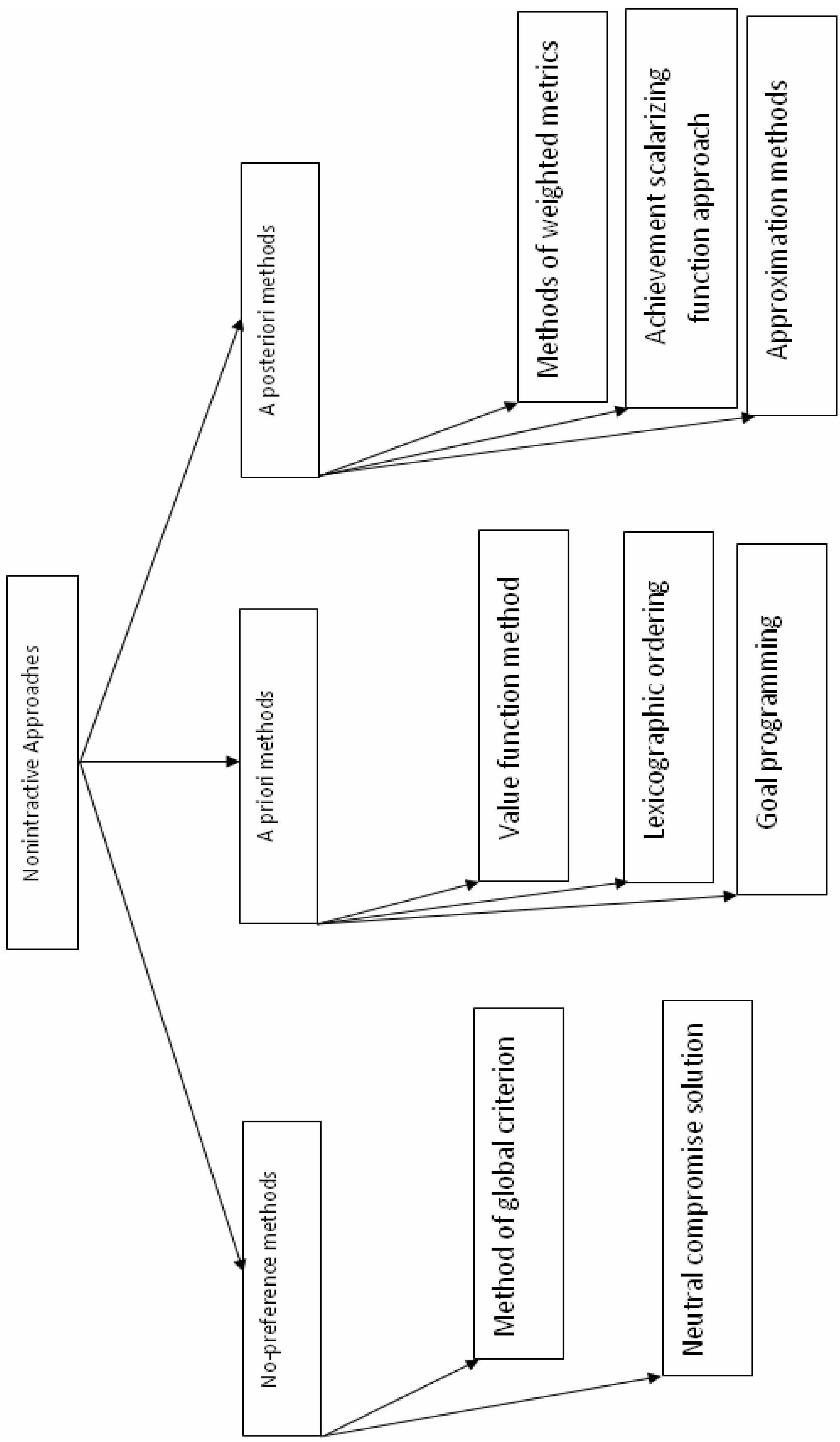
2.2 Pareto Optimality:

In contrast to single-objective optimization, a solution to a multi-objective problem is more of a concept than a definition. Typically, there is no single global solution, and it is often necessary to determine a set of points that all fit a predetermined definition for an optimum. The predominant concept in defining an optimal point is that of Pareto optimality (Pareto 1906), which is defined as follows: Pareto Optimal A point, $x^* \in X$, is Pareto optimal if and only if there does not exist another point, $x \in X$, such that $f_k(x) < f_k(x_k^*)$, and $f_k(x) < f_k(x_k^*)$ for at least one function, (Marler and Arora, 2004).

2.3 Non-interactive Approaches for Solving Multi-Objective Optimization:

Some methods for solving multi-objective optimization problems are classified into the three following classics according to role of a Decision Maker (DM) in the solution process. Sometimes, there is no (DM) and her/his preference information available and in those cases we must use so-called no-preference methods. In all the other classes, the (DM) is assumed to take part in the solution process, (Branke et al., 2008).

The following diagram shows different ways to solve multi-objective problems:



2.3 Basic Methods:

Before we concentrate on the lexicographic method that will be described in section 3, we first mention two well-known methods that can be called basic methods because they are so widely used. Actually, in many applications one can see them being used without necessarily recognizing them as multi-objective optimization methods. In other words, the difference between a modeling and an optimization phase is often blurred and these methods are used in order to convert the problem into a form where one objective function can be optimized with single objective solvers available. The reason for this may be that methods of single objective optimization are more widely known as those of multi-objective optimization. One can say that these two basic methods:-
1) weighting method.
2) ε-constraint method.
Are the ones that first come to one's mind if there is a need to optimize multiple objectives simultaneously, (Branke et al., 2008).

2.4 A Priori Methods:

In a priori methods, the DM must specify her/his preference information (for example, in the form of aspirations or opinions) before the solution process. If the solution obtained is satisfactory, the DM does not have to invest too much time in the solution process. However, unfortunately, the DM does not necessarily know beforehand what it is possible to attain in the problem and how realistic her/his expectations are. In this case, the DM may be disappointed at the solution obtained and may be willing to change one's preference information, (Branke et al., 2008).

3. Lexicographic Method:

3.1 Lexicographic GP problems:

In this section we can use the new technique, namely **Lexicographic Technique**, in solving the multi-objective geometric programming problem which can be defined as:

Find $x = (x_1, \ldots\ldots, x_n)^T$ so as to

$$\min f_{l0}(x) = \sum_{j=1}^{m_{l0}} c_{l0j} \prod_{i=1}^{n} x_i^{a_{l0i}} \qquad l = 1, \ldots\ldots, m \qquad (4)$$

s.t.

$$f_k(x) = \sum_{j=m_{k-1}+1}^{m_k} c_{kj} \prod_{i=1}^{n} x_i^{a_{ji}} \leq 1 \quad k = 1, \ldots\ldots, p \qquad (5)$$

$$x = (x_1, \ldots\ldots, x_n) > 0 \qquad (6)$$

where c_{kj} for all k and j are positive real numbers and a_{l0i} and a a_{ji} are real numbers for all l, j, i, m_{l_0} =number of terms present in the lth objective function. m_k =number of terms present in the kth constraint. In the above multi-objective geometric programming problem there are m number of minimization type objective functions, p number of inequality type constraints and n number of strictly positive decision variables.
Now the above optimization problem can be rewritten as:

$$lex \min\{f_{l0}(x) : x \in R^n, f_k(x) \leq 1; l = 1,2, \ldots\ldots, m, k = 1,2, \ldots p\} \; (7)$$

Equations (4)-(7) form the basis of applying the new technique to the GPP, and hence, we call it as (**Lexicographic GPP**).

3.2 A numerical procedure for solving Lexicographic GPP:

With the lexicographic method, the objective functions are arranged in order of importance. Then, the following optimization problems are solved one at a time:

$$\left.\begin{array}{ll} \min f_l(x) & l = 1, \ldots, m \\ s.t. \quad f_k(x) \leq f_k(x_k^*) & k = 1,2, \ldots\ldots, l-1, l > 1 \end{array}\right\} \quad (8)$$

Here, l represents a function's position in the preferred sequence, and $f_k(x_k^*)$ represents the optimum of the k-th objective function, found in the k-th iteration. After the first iteration (k = 1), $f_k(x_k^*)$is not necessarily the same as the independent minimum of $f_k(x)$ because new constraints have been introduced. The constraints in (8) can be replaced with equalities (Stadler, 1988). Some authors distinguish the hierarchical method from the lexicographic approach. (Waltz, 1967) proposed that the constraints are formulated as $f_k(x) \leq f_k(x_k^*) + \delta_k$. In this case, δ_k are positive tolerances determined by the decision-maker, and as they increase, the feasible region dictated by the objective functions expand. This reduces the sensitivity of the final solution to the initial objective-function ranking process. One may vary δ_k to tighten the constraints and in this way generates different Pareto optimal points.

4. A General Alternative Procedure for Solving Negative Degree of Difficulty(the Construction of the Transformed Problem):

In this section, we analyze the previous multi-objective posynomial geometric programming problem (7) with a negative degree of difficulty $(d < 0)$. Using this we construct a new multi-objective posynomial geometric programming problem named the transformed problem with $n+1$ variables, $p+n+2$, constraints. In this new problem we only minimize the new variable x_{n+1} subject to the same p constraints of the original problem and add one more constraint of the form $f_{l0}(x) \leq x_{n+1}$.

Another constraint is included for each $n+1$ variable in the new problem, with an explicitly indicated lower bound (b_i). The lower bound can be as close to zero as desired and same or different for all variables (two examples are provided in section 6). The set of new constraints is a result of using the strict positivity constraints in the variables (6) in the original problem (4 and 5). They play an important role in the new posynomial problem, providing it with a strictly positive degree of difficulty. Keeping in mind all these considerations, the formulation of the problem constructed is:

$$\min x_{n+1} \tag{9}$$

s.t.

$$f_k(x) = \sum_{j=m_{k-1}+1}^{m_k} c_{kj} \prod_{i=1}^{n} x_i^{a_{ji}} \leq 1 \quad k = 1, \ldots\ldots, p \tag{10}$$

$$f_{p+1}(x) = x_{n+1}^{-1} \sum_{j=m_{k-1}+1}^{m_k} c_{kj} \prod_{i=1}^{n} x_i^{a_{ji}} \leq 1 \tag{11}$$

$$b_i x_i^{-1} \leq 1 \quad i = 1, \ldots\ldots, n+1 \tag{12}$$

where

$c_j > 0, j = 1, \ldots, m;\ a_{ji} \in R, i = 1, \ldots, n, j = 1, \ldots, m, m_p = m,\ m_k > m_{k-1}, \forall\, k.$

For the details see (Allueva et al., 2004).

5. How to Use A General Alternative Procedure for Solving Negative Degree of Difficulty in the Lexicographic GPP:

Let there is $l = 1, \ldots\ldots, m$ objective functions and they must arranged in order of importance; but the difficulty is that the decision maker (DM) must specify his preference information before the solution process. Unfortunately, the (DM) does not necessarily know beforehand what it is possible to attain in the problem and how realistic his expectation are? Some authors try to distinguish the hierarchical method from the lexicographic approach, to reduce the sensitivity of the final solution to the initial objective function ranking process, see (Waltz, 1967), (Osyczka, 1984) and (Rentmeesters et. al., 1996).

5.1: Algorithm (NEW):

Outlines of the new proposed algorithm for solving Lexicographic GPP, with general negative degree of difficulty.

1- Now since there is m number of objective functions, we suggest to solve $p!$ problems, as example if there is bi-objective function, the two solved problems will be:

Problem A	Problem B
$\min f_{10}$ $\min f_{20}$ s.t. $f_k(x) \leq 1 \qquad k = 1, \ldots\ldots, p$	$\min f_{20}$ $\min f_{10}$ s.t. $f_k(x) \leq 1 \qquad k = 1, \ldots, p$

2- Both problem A&B are multi-objective GPP having negative degree of difficulty; the first objective function in problem A and the first objective function in problem B are minimized subject to all original constraints; namely the formulation of those problems are:

Problem A'	Problem B
$\min f_{10}(x)$ s.t. $f_k(x) \leq 1 \qquad k = 1, \ldots\ldots, p$	$\min f_{20}(x)$ s.t. $f_k(x) \leq 1 \qquad k = 1, \ldots\ldots, p$

We note that both problem $A^{'}$ & $B^{'}$ are single objective GP having negative degree of difficulty the general procedure for solving this kind of problems which was mentioned in **Section 4** by (Allueva et. al., 2004):

3- The following new transformed problem gained

Problem $TA^{'}$	Problem $TB^{'}$
$\min x_{n+1}$ s.t. $f_k(x) \le 1$ $f_{p+1}(x) \le 1$ where $f_{p+1}(x) = f_{10}(x)$ $b_i x_i^{-1} \le 1 \quad i = 1, \dots, n+1$	$\min x_{n+1}$ s.t. $f_k(x) \le 1$ $f_{p+1}(x) \le 1$ where $f_{p+1}(x) = f_{20}(x)$ $b_i x_i^{-1} \le 1 \quad i = 1, \dots, n+1$

Note that the problems $TA^{'}$ and $TB^{'}$ are of positive degree of difficulty.

4- Find the exponent matrices of $TA^{'}$ &$TB^{'}$ problems; those matrices must be full rank. For more details of this step see the following notice from (Allueva et. al., 2004):
(*Assume that there is minimum solution to the posynomial GP problems* $A^{'}$&$B^{'}$*,thus, problems* $TA^{'}$&$TB^{'}$ *reach a minimum value that approaches the minimum by an appropriate choice of* b_i) .

5- After making an interface between the GGP lab and Matlab window; use (gpsolve) function to solve the gained problems and find $f_{10}(x^*)$ & $f_{20}(x^*)$ for $TA^{'}$ &$TB^{'}$.

6- In this step the objective function $f_{20}(x)$ is minimized for the problem A" subject to the original constraints with one additional constraint, i.e. $f_{10}(x) \ge f_{10}(x^*)$. Similarly, the objective function $f_{10}(x)$ is minimized for the problem B" subject to the original constraints with one additional constraint, i.e. $f_{20}(x) \ge f_{20}(x^*)$. So the formulation of those two problems will be:

Problem A''	Problem B''
$\min f_{20}(x)$ s.t. $f_k(x) \le 1 \quad k = 1, \dots, p$ $f_{10}(x) \ge f_{10}(x^*)$	$\min f_{10}(x)$ s.t. $f_k(x) \le 1 \quad k = 1, \dots, p$ $f_{20}(x) \ge f_{20}(x^*)$

The two above problems $A'' \,\&\, B''$ are also need to be transformed to change the negative degree of difficulty into positive degree of difficulty.

7- Repeating the step 4 and 5 to get the solution for both problems A''& B' which will be the final solution for problems $A\&B$, respectively.

8- The set of solutions $sol = \begin{Bmatrix} x_1^* \text{ is a solution to problem } A \\ x_2^* \text{ is a solution to problem } B \end{Bmatrix}$ will be regarded as Pareto solutions; note that if we have a GP problem with triple- objective functions, then we will find a set of solutions $sol = \{x_1^*, i = 1, \dots, 6\}$, since $3! = 6$ & so on.

9- Finally we must distinguish if each of those solutions is Pareto optimal; Pareto weakly or Pareto properly solution.

5.2: Note that :- the preference of the (DM) will not decrease but will rather increase if the value of the objective function decreases, while all the other objective values remain unchanged (i.e. less is preferred to move).

6. Numerical Examples:

Example (6.1):

$$\min \quad \begin{matrix} f_1(x) = 3.5\, x_1 x_2^{-2} x_3 x_4^{-0.75} x_8 \\ f_2(x) = 96\, x_1^{-1.8} x_3 x_4^{-0.25} x_7^{\frac{-2}{3}} \end{matrix}$$

s.t.

$288670\, x_1^{-0.875} x_2^{-0.75} + x_3^{-1} x_5^{-0.75} x_7^{0.75} \le 1$

$25819\, x_1^{-0.2} x_3^{-1} x_4^{0.8} + x_6^{-1} x_8^{1.75} \le 1$

$x_1, x_2, \dots\dots, x_8 > 0$

Solution:

The above problem is posynomial GPP and of (-3) degree of difficulty (i.e. $d = 6 - 8 - 1 = -3$).

The matrix of an exponent function is:-

$$a = \begin{bmatrix} 1 & -2 & 1 & -0.75 & 0 & 0 & 0 & 1 \\ -1.8 & 0 & 1 & -0.25 & 0 & 0 & -\frac{2}{3} & 0 \\ -0.875 & -0.75 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & -0.75 & 0 & 0.75 & 0 \\ -0.2 & 0 & -1 & 0.8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1.75 \end{bmatrix}$$

Is full rank (i.e. is of 6 rank). The first problem is

$\min f_1(x) = 3.5\, x_1 x_2^{-2} x_3 x_4^{-0.75} x_8$

s.t.

$288670\, x_1^{-0.875} x_2^{-0.75} + x_3^{-1} x_5^{-0.75} x_7^{0.75} \le 1$

$25819\, x_1^{-0.2} x_3^{-1} x_4^{0.8} + x_6^{-1} x_8^{1.75} \le 1$

$x_1, x_2, \dots\dots, x_8 > 0$.

The degree of difficulty is $d = 5 - 8 - 1 = -4$

the transformed problem is:

$\min x_9$

s.t.

$3.5\ x_1 x_2^{-2} x_3 x_4^{-0.75} x_8\ x_9^{-1} \leq 1$

$288670\ x_1^{-0.875} x_2^{-0.75} + x_3^{-1} x_5^{-0.75} x_7^{0.75} \leq 1$

$25819\ x_1^{-0.2} x_3^{-1} x_4^{0.8} + x_6^{-1} x_8^{1.75} \leq 1$

$b_i x_i^{-1} \leq 1\ ,i = 1,2 \ldots.,9$

where the lower bounds b_i are the appropriate choice and
$f_{l0}(x^*) = x_8^* \qquad l_0 = 1,2$.

As we have seen above, the first objective function $f_1(x)$ was minimized subject to all the original constraints. Now the second objective function $f_2(x)$ will be a minimized subject to the original constraints with one additional constraint i.e. $f_1(x) \geq 3.0422\text{e} - 037$.

$$min\ \ f_2(x) = 96\ x_1^{-1.8} x_3 x_4^{-0.25} x_7^{-2/3}$$

s.t.

$288670\ x_1^{-0.875} x_2^{-0.75} + x_3^{-1} x_5^{-0.75} x_7^{0.75} \leq 1$

$25819\ x_1^{-0.2} x_3^{-1} x_4^{0.8} + x_6^{-1} x_8^{1.75} \leq 1$

$3.5\ x_1 x_2^{-2} x_3 x_4^{-0.75} x_8 \ \geq\ 3.0422\text{e} - 037$
$x_1, x_2, \ldots \ldots, x_8 > 0$

It is clear that the last problem is also of negative degree of difficulty therefore the same above transformer will be needed, and the final results were described in Table (6.1).

Example (6.2):

$$\min \begin{array}{l} f_1(x) = 2.75\ x_1^{-1} x_2 x_3 x_5 x_6^{-0.5} x_9^{-1.5} x_{10}^{-0.25} \\ f_2(x) = \ x_1^{-1} x_3^{-2} x_4^2 x_7^{-0.75} \\ f_3(x) = x_2 x_5^{-0.5} x_6 x_7^{0.75} x_8^{-1} \end{array}$$

s.t.

$5x_1^{-1}x_2x_8^2 \leq 1$

$2.5x_2^{-1}x_3^2x_9^{-3} + 1.5x_3^{-1}x_4^{-0.5}x_{10}^2 \leq 1$

$x_1, x_2, \dots \dots, x_{10} > 0$

The results of this example follow from example (6.1). Hence, we can summarize our numerical results of this example in the following Table (6.2).

Table (6.1)

Case1		Case2	
'x1' 'x2' 'x3' 'x4' 'x5' 'x6' 'x7' 'x8'	[3.1636e+072] [3.0509e+040] [9.3479e+073] [8.5197e+044] [1.2140e+066] [8.7500e+069] [1.9186e+058] [4.4394e+012]	'x1' 'x2' 'x3' 'x4' 'x5' 'x6' 'x7' 'x8'	[4.2245e+040] [1.4530e+073] [4.4722e+068] [1.8280e+053] [5.1419e+057] [5.3322e+078] [9.4022e+033] [2.4501e+028]
$f_1(x)$ $f_2(x)$	4.3766e-029 1.5030e-183	$f_2(x)$ $f_1(x)$	5.8522e-085 1.9110e-213

Table (6.2)

Case1		Case2	
'x1' 'x2' 'x3' 'x4' 'x5' 'x5' 'x7' 'x8' 'x9' 'x10'	[9.5252e+073] [5.2768e+009] [8.7656e+036] [3.3655e+069] [1.5029e+082] [1.7928e+012] [2.8030e+017] [7.3433e+019] [5.8090e+059] [1.5703e+051]	'x1' 'x2' 'x3' 'x4' 'x5' 'x5' 'x7' 'x8' 'x9' 'x10'	[1.6123e+074] [3.1467e+009] [7.9390e+036] [2.4363e+070] [6.3920e+081] [2.8899e+012] [2.1672e+017] [1.0617e+020] [2.2342e+060] [7.5615e+050]
$f_3(x)$ $f_2(x)$ $f_1(x)$	5.3781e-054 1.2704e-022 1.2802e-026	$f_3(x)$ $f_1(x)$ $f_2(x)$	1.0761e-026 5.8151e-021 4.1363e-125
Case3		Case4	
'x1' 'x2' 'x3' 'x4' 'x5' 'x5' 'x7' 'x8' 'x9' 'x10'	[1.5852e+079] [2.4422e+049] [5.1435e+067] [4.5496e+041] [1.6976e+074] [2.0270e+050] [3.2235e+066] [2.3953e+004] [1.8503e+056] [2.8236e+048]	'x1' 'x2' 'x3' 'x4' 'x5' 'x5' 'x7' 'x8' 'x9' 'x10'	[2.8148e+070] [2.7406e+026] [4.5241e+030] [4.3867e+086] [9.1434e+035] [8.5468e+073] [1.6132e+045] [6.6179e+006] [4.2458e+068] [7.0086e+037]
$f_2(x)$ $f_3(x)$ $f_1(x)$	6.4878e-182 1.2067e+108 7.9640e-010	$f_1(x)$ $f_2(x)$ $f_3(x)$	4.7328e-127 2.9795e+109 4.1495e+007
Case5		Case6	
'x1' 'x2' 'x3' 'x4' 'x5' 'x5' 'x7' 'x8' 'x9' 'x10'	[4.5590e+079] [8.6136e+038] [1.8670e+044] [8.9335e+075] [9.6011e+029] [3.6555e+079] [6.2893e+068] [2.2242e+005] [7.8679e+071] [9.7853e+046]	'x1' 'x2' 'x3' 'x4' 'x5' 'x5' 'x7' 'x8' 'x9' 'x10'	[5.4378e+078] [7.0140e+056] [7.1058e+070] [1.1241e+042] [4.1210e+080] [3.2759e+041] [3.1536e+066] [355.2627] [1.1382e+048] [8.6341e+043]
$f_1(x)$ $f_3(x)$ $f_2(x)$	1.2645e-068 5.7378e+149 3.9465e-126	$f_2(x)$ $f_1(x)$ $f_3(x)$	6.1497e-187 1.5504e+026 2.3842e+105

7. Conclusions:

Different GPP methods have been proposed to obtain the optimal solution of the posynomial GP problems with degree of difficulty greater or equal to zero. However, very few articles have been considered with negative degree of difficulties. In general, multi-objective optimization requires more computational effort than single-objective optimization. Unless preferences are irrelevant or completely understood, solution of several single objective problems may be necessary to obtain an acceptable final solution. We don't find any article deals with exactly multi-objective GPP having any negative degree of difficult, therefore we have suggested to combine two well-known approaches and try to formulate $p!$ problems.

References:

1- A. Allueva; J. L. Alejandre and J. M. Gonsalez, A General Alternative Procedure for Solving Negative Degree of Difficulty Problem in Geometric Programming, Springer. (27), (2004), 83-94.

2- D. L. Bricker; J. C. Choi and J. Rajgopal, On Geometric Programming Problems Having Negative Degrees of Difficulty, Journal of operational research, (68), (1993), 427-430.

3- J. Branke; K. Deb; K. Miettinen and R. Słowi´nski, Multi-objective Optimization Interactive and Evolutionary Approaches. Springer-Verlag, Berlin, Heidelberg, (2008).

4- R. J. Duffin; E. L. Peterson and C. M. Zener, Geometric Programming Theory and Application, Wiley, New York, (1967).

5- R.T. Marler and J. S. Arora, Survey of multi-objective optimization methods for engineering. Struct. Multidisc. Optim., (26), (2004), 369–395.

6- F.M. Waltz, An engineering approach: hierarchical optimization criteria. IEEE Trans. Autom. Control AC. (12), (1967), 179–180.