A PROPOSAL ON UTILIZATION OF GAME THEORY TO FIND WEIGHTS FOR MULTIPLE OBJECTIVE LINEAR PROGRAMMING PROBLEMS

İ. Müfit GİRESUNLU - Erhan ÖZDEMİR
Istanbul University, Vocational School of Technical Sciences, Avcılar Campus, Istanbul - TURKEY

Summary : In this paper Multiple Objective Linear Programming (MOLP) problem is transformed into Linear Programming (LP) problem by using suitable weights for each function. Game Theory and Decision Matrix are used to find the weights.

ÇOK AMAÇLI LİNEER PROGLAMLAMA PROBLEMLERİ İÇİN AĞIRLIKLARI BULMADA OYUNLAR KURAMININ KULLANIMINA İLİŞKİN BİR ÖNERİ

Özet : Bu çalışmada Çok Amaçlı Lineer Programlama problemi uygun ağırlıklar kullanılarak Lineer Programlama problemine dönüştürülmektedir. Ağırlıkların bulunmasında Oyunlar Kuramı ve Karar Matrisi kullanılmaktadır.

1. INTRODUCTION

One of the methods for solving MOLP problems is the Weighted-Sums approach. In this approach MOLP problem, which contains more than one objective function, is transformed into LP problem by finding a suitable weight for each objective function. In other words, convex combinations of objective functions are obtained [5]. In view of the fact that the weights cannot be negative and that their sum is equal to one. It is where the difficulty exists to find suitable weights. The difficulty stems from the disparities that exist in the magnitudes of values generated by various objective functions. Moreover, it is usually not possible to find common units of measurement for all objective functions. This is the reason why objective functions are normalized.

In this paper every function is valued at the extreme points of all functions and then the game matrix is constructed using these values [4]. The game matrix entries are normalized in order to find common units of measurement.

Key Words: Game Theory, Linear Programming, Convex Combination.

Solving some problems we need an iteration. Therefore, we construct decision matrix by using differences between the normalized function values for iteration in problems and then the specified weights are calculated from normalized game matrix.

2. WEIGHTED-SUMS APPROACH

The MOLP of the references is

$$\begin{aligned}
&\max \{f_1(\mathbf{x}) = \mathbf{C}^1 \mathbf{x}\} \\
&\max \{f_2(\mathbf{x}) = \mathbf{C}^2 \mathbf{x}\} \\
&\quad \vdots \\
&\max \{f_k(\mathbf{x}) = \mathbf{C}^k \mathbf{x}\} \\
&\text{s.t. } x \in \mathbf{S}
\end{aligned}$$

where

$$\mathbf{S} = \{\mathbf{x} \mid \mathbf{x} \in \mathbf{R}^n,\ \mathbf{A}\mathbf{x} = \mathbf{b},\ \mathbf{x} \geqslant \mathbf{0},\ \mathbf{b} \in \mathbf{R}^m\}$$

or

$$\max \{F(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x})] \mid \mathbf{x} \in \mathbf{S}\}. \tag{1}$$

The intended purpose of problem (1) is to find $\mathbf{x}^* \in \mathbf{S}$ efficient vector (efficient solution) which maximizes all of the objective functions simultaneously. Namely, $\mathbf{x}^* \in \mathbf{S}$ is efficient iff there does not exist another $\mathbf{x} \in \mathbf{S}$ such that $F(\mathbf{x}) \geqslant F(\mathbf{x}^*)$ and $F(\mathbf{x}) \neq F(\mathbf{x}^*)$. Otherwise $\mathbf{x}^*$ is inefficient [3]. Therefore MOLP problems are also called vector maximum problems [2].

Weighted-Sums approach mentioned above is the method that is mostly used to solve problem (1). In this approach $\mathbf{A}$ denotes the set of weighting vectors where

$$\mathbf{A} = \left\{\lambda \mid \lambda \in \mathbf{R}^k,\ \lambda_i \geq 0, \sum_{i=1}^{k} \lambda_i = 1\right\}.$$

Problem (1) is transformed into LP such that

$$\max \left\{f_{k+1}(\mathbf{x}) = \sum_{i=1}^{k} \lambda_i f_i(\mathbf{x}) \mid \mathbf{x} \in \mathbf{S}\right\}. \tag{2}$$

Now let us give the following theorem, without proof, to provide the relationship between problems (1) and (2), [6].

Theorem. $\mathbf{x}^* \in \mathbf{S}$ is efficient $\Leftrightarrow \exists\, \lambda \in \mathbf{A}$ such that $\mathbf{x}^*$ maximizes the weighted-sums LP (2).

3. FINDING THE WEIGHTS

Belenson and Kapur [1] have transformed (1) into (2) with two person-zero sum game theory. In [1] they have computed λ_i weights using game matrix and obtained weighted-sums of functions.

In this paper, we follow same procedure and obtain $\mathbf{x}_i^*$ extreme points and corresponding function values maximizing all $f_i(\mathbf{x}_i)$ on $\mathbf{S}$. Game matrix $\mathbf{G}$ is formed by using these values. In $\mathbf{G}$, for all $i = 1,2,\dots,k$

$$f_i(\mathbf{x}_i^*) \geqslant f_i(\mathbf{x}_j^*) \ ; \quad j = 1,2,\dots,k. \tag{3}$$

The values satisfying inequality (3) are denoted in $\mathbf{G}$ as follows:

$$f_i(x_j^*) = f_{ij} \ ; \quad j = 1,2,\dots,k.$$

The game matrix $\mathbf{G}$ is as shown in Table 1.

$$G = \begin{array}{c|cccccc}
 & \mathbf{x}_1^* & \mathbf{x}_2^* & \cdots & \mathbf{x}_j^* & \cdots & \mathbf{x}_k^* \\
\hline
f_1 & f_{11} & f_{12} & \cdots & f_{1j} & \cdots & f_{1k} \\
f_2 & f_{21} & f_{22} & \cdots & f_{2j} & \cdots & f_{2k} \\
\vdots & \vdots & \vdots & & \vdots & & \vdots \\
f_i & f_{i1} & f_{i2} & \cdots & f_{ij} & \cdots & f_{ik} \\
\vdots & \vdots & \vdots & & \vdots & & \vdots \\
f_k & f_{k1} & f_{k2} & \cdots & f_{kj} & \cdots & f_{kk}
\end{array}$$

Table 1: Game (Pay-off) matrix

Belenson and Kapur have normalized among all elements in rows of matrix $\mathbf{G}$. In general, normalization can be accomplished as follows :

Let $\mathbf{G}_N$ represent the normalized game matrix $\mathbf{G}$, as shown in Table 2. $\mathbf{G}_N$ matrix entries z_{ij} can be formed by

$$z_{ij} = \frac{f_{ij}}{f_{ii}} \ ; \quad j = 1,2,\dots,k. \tag{4}$$

$$\mathbf{G}_N = \begin{array}{c|cccccc}
 & \mathbf{x}_1^* & \mathbf{x}_2^* & \cdots & \mathbf{x}_j^* & \cdots & \mathbf{x}_k^* \\
\hline
f_1 & 1 & z_{12} & \cdots & z_{1j} & \cdots & z_{1k} \\
f_2 & z_{21} & 1 & \cdots & z_{2j} & \cdots & z_{2k} \\
\vdots & \vdots & \vdots & & \vdots & & \vdots \\
f_i & z_{i1} & z_{i2} & \cdots & 1 & \cdots & z_{ik} \\
\vdots & \vdots & \vdots & & \vdots & & \vdots \\
f_k & z_{k1} & z_{k2} & \cdots & z_{kj} & \cdots & 1
\end{array}$$

Table 2: Normalized Game Matrix

So, $\mathbf{G}_N$ entries represent the relative expectancy of each objective function with respect to its maximum value.

We calculate λ_i' weights (strategies) on normalized matrix $\mathbf{G}_N$ by LP. But these λ_i weights are not $\mathbf{G}$ matrix weights. The following calculations are required in order to find optimum λ_i^* weights of game matrix $\mathbf{G}$:

$$m_i = \frac{\lambda_i'}{f_{ii}}, \quad M = \sum_{i=1}^{k} m_i, \; \lambda_i' = \frac{m_i}{M}; \quad i = 1,2,\ldots,k. \tag{5}$$

There may be a problem in the normalization process all $f_{ij} < 0$ or $= 0$. In that case, a sufficiently large fixed constant K must be added to all entries in the matrix $\mathbf{G}$.

4. THE PROPOSED METHOD FOR FINDING THE NEW WEIGHTS

By using λ_i^* weights, we have composite function

$$f_{k+1} = \sum_{i=1}^{k} \lambda_i^* f_i \,.$$

This function is solved by LP on S and efficient point $\mathbf{x}_{k+1}^*$ thus obtained. $\mathbf{x}_{k+1}^*$ is the efficient point of problem (1) as well, according to the theorem given in Section 2 above. The optimum solution vector corresponding to x_{k+1} is

$$F^* = [f_1(\mathbf{x}_{k+1}^*), \ldots, f_k(\mathbf{x}_{k+1}^*)]. \tag{6}$$

In case the decision-maker does not find solution (5) to be satisfactory, game matrix $\mathbf{G}$ is rearranged by using extreme point of f_{k+1} function.

In Belenson and Kapur's work, the extreme point which is replaced by $\mathbf{x}_{k+1}^*$ is selected as

$$f_\alpha(\mathbf{x}_{k+2}^*) < f_\alpha(\mathbf{x}_{k+1}^*)$$

and

$$f_i(\mathbf{x}_{k+2}^*) < f_i(\mathbf{x}_{k+1}^*), \quad i \neq \alpha$$

for at least one i, where $f_\alpha(\mathbf{x})$ is the least preferred objective function for the decision-maker. Instead of this, we propose a different and more simplified method of selection where decision matrix $\mathbf{D}$ is used.

In our decision-matrix the entries are calculated as follows:

$$d_{ij} = \begin{cases} 1 - z_{ij}, & i \neq j \\ 0, & i = j. \end{cases} \tag{7}$$

The decision matrix **D** is as shown in Table 3.

$$
\mathbf{D} = \begin{array}{c|cccccc}
 & \mathbf{x}_1^* & \mathbf{x}_2^* & \dots & \mathbf{x}_i^* & \dots & \mathbf{x}_k^* \\
\hline
f_1 & 0 & d_{12} & \dots & d_{1i} & \dots & d_{1k} \\
f_2 & d_{21} & 0 & \dots & d_{2i} & \dots & d_{2k} \\
\vdots & \vdots & \vdots & & \vdots & & \vdots \\
f_i & d_{i1} & d_{i2} & \dots & 0 & \dots & d_{ik} \\
\vdots & \vdots & \vdots & & \vdots & & \vdots \\
f_k & d_{k1} & d_{k2} & \dots & d_{ki} & \dots & 0
\end{array}
$$

Table 3: Decision Matrix

From **D**,

$$\max_i [\max_j d_{ij}] = d_{ij}^* \; ; \quad i, j = 1,2, \dots, k \tag{8}$$

is determined. The extreme point x_i^* which corresponds to d_{ij}^* is replaced by $\mathbf{x}_{k+1}^*$ which is the extreme point of f_{k+1} and the new game matrix is found. The same process outlined in Section 3 is repeated and new composite function is found and so on.

The process of finding the new solution vector comes to a halt when the new extreme point of last composite function is equal to one of the old extreme points which already exist. So, the optimum solution is equal to previous solution.

5. EXAMPLES

1. $\max\, [f_1(\mathbf{x}) = -x_1]$

   $\max\, [f_2(\mathbf{x}) = 0.1\ x_1 + 0.2\ x_2]$

   s.t.

$$
\begin{aligned}
-x_1 + x_2 &\leqslant 1 \\
x_1 + x_2 &\leqslant 7 \\
x_1 + x_2 &\geqslant 1 \\
x_1 \qquad &\leqslant 5 \\
x_2 &\leqslant 3 \\
x_1\ ,\ \ x_2 &\geqslant 0.
\end{aligned}
$$

If each function given above is solved by LP, the following extreme points are obtained:

$$x_1^* = (0,1)\ , \quad x_2^* = (4,3).$$

Hence the matrices $\mathbf{G}$, $\mathbf{G}_N$ and $\mathbf{D}$ are:

$$\mathbf{G} = \begin{array}{c|cc} & \mathbf{x}_1^* & \mathbf{x}_2^* \\ \hline f_1 & 0 & -4 \\ f_2 & 0.2 & 1 \end{array} \overset{K=4}{\Longrightarrow} \mathbf{G}_N = \begin{array}{c|cc} & \mathbf{x}_1^* & \mathbf{x}_2^* \\ \hline f_1 & 1 & 0 \\ f_2 & 4.2/5 & 1 \end{array} \Longrightarrow \mathbf{D} = \begin{array}{c|cc} & \mathbf{x}_1^* & \mathbf{x}_2^* \\ \hline f_2 & 0 & 1 \\ f_2 & 0.8/5 & 0 \end{array}.$$

Solve for λ' using LP for $\mathbf{G}_N$ and then calculate m.

$$\lambda' = (4/29,\ 25/29),\ \mathbf{m} = (1/29,\ 5/29),\ M = 6/29 \Longrightarrow \lambda^* = (1/6,\ 5/6) \Longrightarrow$$
$$f_3(\mathbf{x}) = (-1/12)\ x_1 + (1/6)\ x_2\,.$$

Therefore $x_3^* = (2,3)$ and $F^* = (-2,\ 0.8)$.

From $\mathbf{D}$, $d_{ij}^* = 1$, so $\mathbf{x}_2^* = (4,3)$ is replaced by $\mathbf{x}_3^* = (2,3)$ and the new game matrix is

$$\mathbf{G} = \begin{array}{c|cc} & x_1^* & x_2^* \\ \hline f_1 & 0 & -2 \\ f_2 & 0.2 & 0.8 \end{array} \overset{K=2}{\Longrightarrow} \mathbf{G}_N = \begin{array}{c|cc} & x_1^* & x_2^* \\ \hline f_1 & 1 & 0 \\ f_2 & 2.2/2.8 & 1 \end{array}.$$

From $\mathbf{G}_N$ we have

$$\lambda' = (0.83,\ 0.17),\ \mathbf{m} = (0.42,\ 0.06),\ M = 0.48 \Longrightarrow \lambda^* = (0.88,\ 0.12) \Longrightarrow$$
$$f_1(\mathbf{x}) = -0.87\ x_1 + 0.24\ x_2\,.$$

Therefore $\mathbf{x}_4^* = (0,1) \Longrightarrow \mathbf{x}_4^* = \mathbf{x}_1^*$. This means that this iteration is unnecessary.

Optimum solution: $\mathbf{x}_3^* = (2,3)$ and $F^* = (-2, 0.8)$.

2. $\max\,[f_1(\mathbf{x}) = -2x_1 + x_2]$

$$\max\,[f_2(\mathbf{x}) = 3x_1 - x_2]$$
$$\max\,[f_3(\mathbf{x}) = -3x_1 + 4x_2]$$

s.t.

$$\begin{aligned} -x_1 + x_2 &\leqslant 1 \\ x_1 + x_2 &\leqslant 7 \\ x_1 \quad\ &\leqslant 5 \\ x_2 &\leqslant 3 \\ x_1\,,\ x_2 &\geqslant 0. \end{aligned}$$

If each function given above is solved by LP, the following extreme points are obtained:

$$\mathbf{x}_1^* = (0,1),\quad \mathbf{x}_2^* = (5,0)\quad \mathbf{x}_3^* = (2,3).$$

Hence the matrices **G**, $\mathbf{G}_N$ and **D** are:

$$G = \begin{array}{c|ccc} & \mathbf{x}_1^* & \mathbf{x}_2^* & \mathbf{x}_3^* \\ \hline f_1 & 1 & -10 & -1 \\ f_2 & -i & 15 & 3 \\ f_3 & 4 & -15 & 6 \end{array} \Rightarrow \mathbf{G}_N = \begin{array}{c|ccc} & \mathbf{x}_1^* & \mathbf{x}_2^* & \mathbf{x}_3^* \\ \hline f_1 & 1 & -10 & -1 \\ f_3 & -1/15 & 1 & 1/5 \\ f_3 & 2/3 & -5/2 & 1 \end{array}$$

$$\Rightarrow \mathbf{D} = \begin{array}{c|ccc} & \mathbf{x}_1^* & \mathbf{x}_2^* & \mathbf{x}_3^* \\ \hline f_1 & 0 & 11 & 2 \\ f_2 & 16/15 & 0 & 4/5 \\ f_3 & 1/3 & 7/2 & 0 \end{array}$$

Solve for λ' using LP for $\mathbf{G}_N$ and then calculate m.

$$\lambda' = (0, 0.75, 0.25),\ \mathbf{m} = (0, 0.05, 0.04),\ M = 0.09 \Rightarrow \lambda^* = (0, 0.54, 0.46)$$

$$\Rightarrow f_4(\mathbf{x}) = 0.26\, x_1 + 1.29\, x_2.$$

Therefore $\mathbf{x}_4^* = (4,3)$ and $F^* = (-5, 9, 0)$. From **D**, $d_{ij}^* = 11$, so $\mathbf{x}_2^* = (5,0)$ is replaced by $\mathbf{x}_4^* = (4,3)$. If we continue one more iteration we find $\mathbf{x}_5^* = (2,3)$ extreme point. Therefore $\mathbf{x}_5^* = (2,3) \Rightarrow \mathbf{x}_5^* = \mathbf{x}_2^*$. This means that this iteration is unnecessary.

Optimum solution: $\mathbf{x}_4^* = (4,3)$ and $F^* = (-5, 9, 0)$.

6. CONCLUSION

In this paper, for given multi-criterion linear programming problem, a composite function has been obtained by finding the relative optimal weights through a two person zero-sum a game matrix with mixed strategies. In doing this, the efficient points of k functions have been found by LP and the game matrix of $k \times k$ with function values of k efficient points.

When we find the $k+1$'th composite function, the first k functions are to be comparable in order that $k+1$'th composite function is meaningful. Normalization process among functions has therefore been introduced in the literature. In this paper, decision matrix **D** is formed on the matrix $\mathbf{G}_N$ by taking the differences between the value of each function at it's own extreme point and the values of the same function at the remaining $k-1$ extreme points.

The process of finding the new solution vector comes to a halt when the extreme point $\mathbf{x}_{k+1}^*$ is equal to one of the old extreme points which already exist. If some functions have alternative solutions, these solutions are eliminated by d_{ij}^* which is selected from **D**.

REFERENCES

[1] BELENSON, S.M. and KAPUR, K.C. : *An Algorithm for Solving Multicriterion Linear Programming Problems With Examples,* Operational Research Quarterly, 24 (1973), No. I, 65-77.

[2] GEOFFRION, A.M. : *Proper Efficiency and the Theory of Vector Maximization,* Journal of Mathematical Analysis and Applications, 22 (1968), 618-630.

[3] ISERMANN, H. : *Proper Efficiency and the Linear Vector Maximization Problem,* Operations Research, 22 (1974), 189-191.

[4] Mc KINSEY, J. : **Introduction to the Theory of Games**, McGraw - Hill Book Company, Inc., 1952, pg. 24.

[5] ROBERTS, A.W. and VARBERG, D.E. : **Convex Functions**, Academic Press, New York and London, 1973, pg. 75.

[6] STEUER, R.E. : **Multiple Criteria Optimization: Theory, Computation and the Applications**, John Wiley and Sons, 1986, pg. 215.