

# Diagnosis and Resolution of Infeasibility in the Constraint Method for Solving Multi Objective Linear Programming Problems

Mohammadreza Safi, Hossein Zare Marzooni

Department of Mathematics, Semnan University, Semnan, Iran  
 Email: safi\_mohammadreza@yahoo.com, hossein.zare70@gmail.com

Received September 18, 2011; revised October 20, 2011; accepted November 5, 2011

## ABSTRACT

In this paper we discuss about infeasibility diagnosis and infeasibility resolution, when the constraint method is used for solving multi objective linear programming problems. We propose an algorithm for resolution of infeasibility, which is a combination of interactive, weighting and constraint methods. Numerical examples are provided to illustrate the techniques developed.

**Keywords:** Multi Objective Linear Programming; Weighting Method; Constraint Method; Infeasibility Analysis; IIS

## 1. Introduction

Almost every important real world problem involves more than one objective. The multi objective linear programming (MOLP) problem can be formulated as follows:

$$\begin{aligned} \max z &= (c_1^T x, \dots, c_k^T x) \\ s.t. x &\in X = \{x : Ax \leq b, x \geq 0\} \end{aligned} \quad (1)$$

where  $c_j^T, j = 1, \dots, k$  are n-dimensional vectors,  $x$  is an n-dimensional vector,  $b$  is an m-dimensional vector and  $A$  is an  $m \times n$  matrix.

Excluding the trivial case in which a point exists in the feasible region which maximizes all objectives simultaneously, we must often propose a compromise solution to decision maker (DM). In the special case, the point that simultaneously maximizes all objectives is called complete optimal solution. In general, such point rarely exists. Thus, instead of complete optimal solution, Pareto optimal solution (POS) is introduced.  $x^*$  is a POS for Problem (1.1) if there does not exist another  $x$  such that  $c_i^T x \geq c_i^T x^*$  for all  $i$  and  $c_l^T x > c_l^T x^*$  for at least one  $1 \leq l \leq k$ .

There are several methods for solving the MOLP Problem (1). By the utility function method [1] we obtain a compromise solution. The weighting method proposed by Kuhn and Tucker [2], the constraint method proposed by Haimes *et al.* [3] and the weighted minimax method proposed by Bowman [4], characterize POSs. Since then, many different approaches are developed for obtaining POSs and for improving the previous methods [5-7].

Also, there are several fuzzy approaches for solving MOLP problems such as [7-9].

In all of the methods and approaches decision maker (DM) has an essential role. However, choosing unsuitable bounds in the constraint method by DM may occur infeasibility in the problem.

Many published researches have discussed diagnosis of infeasibility, [1,10-13], however, there are few papers deal with infeasibility resolution [14-16]. One of the useful references is [17].

In this paper we use the deletion filter and the elastic filter algorithms for isolating IIS (*irreducible infeasible subsystems*) in order to diagnose infeasibility. For resolution of infeasibility and obtaining a POS, we propose an algorithm which is a combination of interactive, weighting and constraint methods. Also we recall the fuzzy method of Leon and Liern [14] for repairing the constraint, in order to attain a feasible space.

Section 2 recalls the constraint method for obtaining a POS for the multi objective linear programming problem. Section 3 discusses about infeasibility analysis and recall two useful filters for isolating IIS. The resolution of infeasibility is discussed in Section 4. When the smallest cardinality set of constraints to cover all IISs is singleton, we have a special case of infeasibility. An interactive approach for this case is proposed in Section 4. For the other cases we propose a combination of the weighting method and the constraint method. Finally we recall the approach of Leon and Liern for repairing infeasibility. Numerical examples are provided to illustrate the technique developed in this chapter.

## 2. Solving MOLP Problems

There are several methods for solving MOLP problems and obtaining the POSs [1]. Among the many possible ways of scalarizing the MOLP, the weighting and the constraint methods are the most famous methods. In these methods DM should determine the weights and the bounds for every objective, respectively.

In this section we recall the constraint method. However, determining good weights and suitable bounds is a problem for DM. Therefore, we propose a combination method of the weighting and the constraint method.

### 2.1. The Constraint Method

The constraint method for characterizing POS is to solve the constraint problem formulated by taking one objective as the objective function and letting all the other objective functions be inequality constraints. The constraint problem is defined by

$$\begin{aligned} & \max \mathbf{c}_j^T \mathbf{x} \\ & \text{s.t. } \mathbf{c}_i^T \mathbf{x} \geq \varepsilon_i, i = 1, \dots, k : i \neq j \\ & \mathbf{x} \in \mathbf{X}, \end{aligned} \quad (2)$$

where  $\varepsilon_i, i = 1, \dots, k; i \neq j$ ; are chosen by DM.

If  $\mathbf{x}^* \in \mathbf{X}$  is a unique optimal solution to Problem (2) for some  $\varepsilon_i, i = 1, \dots, k; i \neq j$  then  $\mathbf{x}^*$  is a POS to Problem (1). Conversely, If  $\mathbf{x}^* \in \mathbf{X}$  is a POS of Problem (1), then  $\mathbf{x}^*$  is an optimal solution of the Problem (2) for some  $\varepsilon_i, i = 1, \dots, k; i \neq j$ .

Set  $\mathbf{S} = \{ \mathbf{x} : \mathbf{c}_i^T \mathbf{x} \geq \varepsilon_i, i = 1, \dots, k : i \neq j, \mathbf{x} \geq 0 \}$ . Choosing unsuitable  $\varepsilon_i$  by DM may occur some inconsistency in the constraints and its consequent is  $\mathbf{S} = \emptyset$  or  $\mathbf{S} \cap \mathbf{X} = \emptyset$  even though  $\mathbf{S} \neq \emptyset$  and  $\mathbf{X} \neq \emptyset$ . This inconsistency follows the infeasibility in the problem.

Here for focus on the objectives, we discuss only the case  $\mathbf{S} \neq \emptyset$ , and suppose that, after repairing  $\mathbf{S}$  and attaining  $\mathbf{S} \neq \emptyset$  we have  $\mathbf{S} \cap \mathbf{X} \neq \emptyset$ .

### 2.2. A Combination Method

Sometimes, DM has suitable bounds for some objectives, tentatively, but for the other objectives he or she hasn't any assessment. Similarly, in weighting method, it's possible, DM has some weights for comparison of some objectives together, but he or she doesn't know how to appropriate some weights for the other objectives. In this case the combination of the two methods may be useful.

Consider Problem (1). Suppose DM has some good weights  $w_i > 0, i \in J_1$  for comparison of  $\mathbf{c}_i^T \mathbf{x}, i \in J_1$  and some suitable lower bounds  $\varepsilon_i, i \in J_2$  for  $\mathbf{c}_i^T \mathbf{x}, i \in J_2$ , where  $\{J_1, J_2\}$  is a partition for  $\{1, 2, \dots, k\}$ . We solve the following problem in order to obtain a POS:

$$\max z = \sum_{i \in J_1} w_i \mathbf{c}_i^T \mathbf{x}$$

$$\text{s.t. } \mathbf{c}_i^T \mathbf{x} \geq \varepsilon_i, i \in J_2. \quad (3)$$

$$\mathbf{x} \in \mathbf{X}$$

**Theorem 2.1.** *If Problem (3) has a unique optimal solution  $\mathbf{x}^*$ , then  $\mathbf{x}^*$  is a POS for Problem (1).*

**Proof:** *Let (contrary)  $\mathbf{x}^*$  be inefficient, then there exist  $\bar{\mathbf{x}} \in \mathbf{X}$ , such that  $\mathbf{c}_i^T \bar{\mathbf{x}} \geq \mathbf{c}_i^T \mathbf{x}^*$  for all  $i \in J_1$ ;  $\mathbf{c}_i^T \bar{\mathbf{x}} > \mathbf{c}_i^T \mathbf{x}^*$ , for all  $i \in J_2$ ;  $\mathbf{c}_i^T \bar{\mathbf{x}} > \mathbf{c}_i^T \mathbf{x}^*$ , for some  $i \in J_1$  or for some  $i \in J_2$ . Thus*

$$\sum_{i \in J_1} w_i \mathbf{c}_i^T \bar{\mathbf{x}} \geq \sum_{i \in J_1} w_i \mathbf{c}_i^T \mathbf{x}^*,$$

*contradiction to unique optimality of  $\mathbf{x}^*$ . □*

## 3. Diagnosis of the Infeasibility

All infeasible systems have one or more *irreducible infeasible subsystems (IISs)* of constraints [18]. An IIS has the property that it is itself infeasible, but any proper subsystem is feasible. An infeasible set of constraints can be rendered feasible by deleting or repairing at least one member of every IIS it contains. Finding the smallest cardinality set of constraints to cover all IISs is known as the minimum-cardinality IIS set-covering problem (MIN IIS COVER) [19].

There are several practical issues related to IIS isolation. An excellent summary of the recently developed algorithmic methods has appeared in [17]. Here we recall *the deletion filter* and *the elastic filter*. The deletion filter guarantees the identification of exactly one IIS, but the exiting of the elastic filter is a small infeasible set that is not necessarily an IIS, but it has at least one IIS.

If the number of objectives in Problem (1) is small, it's better we use deletion filter. Else, we use elastic filter and then use deletion filter on the exiting of elastic filter for obtaining an IIS.

It may be that there are multiple infeasibilities in the model, hence IIS isolation typically is used in a cyclic manner

- 1) *Isolate an IIS;*
- 2) *Determine a repair for this IIS;*
- 3) *If the model is still infeasible, go to step 1).*

### 3.1. The Deletion Filter

The deletion filter proposed by Chinneck and Dravnieks [18] guarantees the identification of exactly one IIS after a single pass through the set of constraints. This is an essential property possessed by very few of the IIS isolation methods. In the following algorithm, for diagnosis infeasibility of the problems, we use *the phase I* of the simplex method.

**INPUT:** *an infeasible set of constraints.*

*For each constraint in the set:*

- 1) *Temporarily drop the constraint from the set.*
- 2) *Test the feasibility of the reduced set.*

*IF feasible THEN return dropped constraint to the set.*

*ELSE (infeasible) drop the constraint permanently.*

**OUTPUT:** constraints constituting a single IIS.

**Theorem 3.1:** *The deletion filter returns exactly one IIS.*

*Proof:* See [18].

### 3.2. The Elastic Filter

The method originally described by Brown and Graves [20]. A fully elastic program adds nonnegative elastic variables  $e_i$  to every constraint and minimizes the summation of these variables as an elastic objective function. This allows finding a feasible solution for the original infeasible model. Namely we solve the following problem:

$$\begin{aligned} \min e &= \sum_{\substack{i=1 \\ i \neq j}}^k e_i \\ \text{s.t. } \mathbf{c}_i^T \mathbf{x} + e_i &\geq \epsilon_i, i = 1, \dots, k; i \neq j, \\ \mathbf{x} &\geq 0, e_i \geq 0, i = 1, \dots, k; i \neq j. \end{aligned} \quad (4)$$

The details of the algorithm are shown in the following:

**INPUT:** an infeasible set of constraints.

- 1) Make all constraints elastic by incorporating non-negative elastic variables  $e_i$ .
- 2) Solve the model using the elastic objective function.
- 3) **IF feasible THEN** enforce the constraints in which any  $e_i > 0$  by permanently removing, Go to step 2.

**ELSE (infeasible) Exit.**

**OUTPUT:** the set of de-elasticized enforced constraints contains at least one IIS.

### 4. Resolution of the Infeasibility

The second major aspect of infeasibility analysis is the infeasibility resolution to repair the problem to make it feasible. However, most published researches and practice results in recent years have focused on the diagnosis side. Little investigation has been made in infeasibility resolution [13,14,16].

#### 4.1. Resolution of Infeasibility in the Constraint Method

The smallest cardinality set of constraints to cover all IISs is known as the minimum-cardinality IIS set-covering problem (MIN IIS COVER). There are several algorithms for finding MIN IIS COVER [17].

In this paper we concentrate on infeasibility in constraint objectives. In almost all of real problems, the number of objectives is very less than the number of constraints. It is usual that in many problems, MIN IIS

COVER for the set of constraint objectives be singleton. For this special case our algorithm proposes an interactive method, and for the other cases we use the combination method mentioned in Section 2.2 or the approach of Leon and Liern (Section 4.2) in order to repair the constraints and resolve infeasibility. Besides, a POS is obtained.

#### The Combination Algorithm

Consider Problem (2):

- 1) Solve the Problem, (4):
- 2) Set  $I_1 = \{i; e_i = 0\}$ ,  $I_2 = \{i; e_i \neq 0\}$ ,
- 3) **IF**  $I_2 = \emptyset$ , there is no infeasibility, solve Problem (2) and STOP.
- 4) **IF**  $I_2 = \{t\}$  for some  $1 \leq t \leq m, t \neq j$ , and  $t = w$ , then, however, MIN IIS COVER is singleton, but there is a cycling in the algorithm. Use Leon and Liern method for solving Problem (2).

**Else, IF**  $I_2 = \{t\}$  for some  $1 \leq t \leq m, t \neq j$ , and  $t \neq w$ , then MIN IIS COVER is singleton, set  $w = t$ , replace  $\mathbf{c}_j^T \mathbf{x}$  by  $\mathbf{c}_t^T \mathbf{x}$  in Problem (2), take  $\epsilon_j$  from DM, change the role of  $j$  and  $t$  in Problem (4) and go to Step 1.

5) **IF**  $|I_2| \geq 2$  and  $I_1 \neq \emptyset$  then take  $w_i, i \in I_2 \cup \{j\}$  from DM and solve Problem (3), STOP.

6) **IF**  $I_1 = \emptyset$ , then use Leon and Liern method for solving Problem (2).

**Example 4.1** Consider the following problem:

$$\begin{aligned} \min & 3x_1 + 3x_2 + 3x_3 \\ \min & 2x_1 + x_2 + 2x_3 \\ \min & 4x_1 + 4x_2 + 2x_3 \\ \min & x_1 + 6x_2 + 5x_3 \\ \text{s.t. } & 4x_1 + 2x_2 + 4x_3 \geq 18 \\ & x_1, x_2, x_3 \geq 0. \end{aligned}$$

Let DM chooses the first objective as the main and  $\epsilon_2 = 18, \epsilon_3 = 6, \epsilon_4 = 14$  as the upper bound of the other objectives. So the problem of finding a POS changes to solving the following problem:

$$\begin{aligned} \min & 3x_1 + 3x_2 + 3x_3 \\ \text{s.t. } & 2x_1 + x_2 + 2x_3 \leq 18 \\ & 4x_1 + 4x_2 + 2x_3 \leq 6 \\ & x_1 + 6x_2 + 5x_3 \leq 24 \\ & 4x_1 + 2x_2 + 4x_3 \geq 18 \\ & x_1, x_2, x_3 \geq 0. \end{aligned}$$

The last problem is infeasible. By solving Problem (4) we obtain  $e_2^* = e_4^* = 0$ , but  $e_3^* \neq 0$ . Namely MIN IIS COVER is singleton. Thus we take  $4x_1 + 4x_2 + 2x_3 \leq 14$  as the main objective. Suppose that DM chooses  $\epsilon_1 = 14$ . The new problem is

$$\begin{aligned} & \min 4x_1 + 4x_2 + 2x_3 \\ & s.t. 2x_1 + x_2 + 2x_3 \leq 18 \\ & 3x_1 + 3x_2 + 3x_3 \leq 14 \\ & x_1 + 6x_2 + 5x_3 \leq 24 \\ & 4x_1 + 2x_2 + 4x_3 \geq 18 \\ & x_1, x_2, x_3 \geq 0. \end{aligned}$$

The last problem has the optimal solution  $(x_1^*, x_2^*, x_3^*) = (0, 0, 4.5)$  that is a POS for the main problem.  $\square$

**Example 4.2** Consider the following problem:

$$\begin{aligned} & \min 3x_1 + 3x_2 + 3x_3 \\ & \min 2x_1 + 3x_2 + 2x_3 \\ & \min 3x_1 - 2x_2 + 4x_3 \\ & \max x_1 + 2x_2 + x_3 \\ & \max x_1 + 4x_2 + x_3 \\ & s.t. 2x_1 + 3x_2 + 4x_3 \geq 2 \\ & x_1, x_2, x_3 \geq 0. \end{aligned}$$

Let DM choose the first objective as the main and  $\varepsilon_2 = 6, \varepsilon_3 = 6, \varepsilon_4 = 8, \varepsilon_5 = 9$  as the bounds of the other objectives.

Solving Problem (3.1) leads to  $(x_1^*, x_2^*, x_3^*) = (0, 2.25, 0)$  and  $(\varepsilon_2^*, \varepsilon_3^*, \varepsilon_4^*, \varepsilon_5^*) = (0.75, 0, 3.5, 0)$ . Thus,  $I_1 = \{2, 4\}$  and  $I_2 = \{1, 3\}$  i.e.  $I_1 \neq \emptyset$  and  $|I_2| \geq 2$  according to our algorithm, we must solve Problem (2.2). Let DM choose  $w_1 = 0.5, w_2 = 0.2, w_4 = 0.3$ . Therefore, we have the following numerical form of Problem (2.2):

$$\begin{aligned} & \min 0.5(3x_1 + 3x_2 + 3x_3) + 0.2(2x_1 + 3x_2 + 2x_3) \\ & \quad + 0.3(-x_1 - 2x_2 - x_3) \\ & s.t. 3x_1 - 2x_2 + 4x_3 \leq 6 \\ & x_1 + 4x_2 + x_3 \geq 9 \\ & 2x_1 + 3x_2 + 4x_3 \geq 2 \\ & x_1, x_2, x_3 \geq 0. \end{aligned}$$

The optimal solution is again  $(x_1^*, x_2^*, x_3^*) = (0, 2.25, 0)$ .  $\square$

### 4.2. Resolution of Infeasibility by Fuzzy Approach

In this section we recall the fuzzy method proposed by Leon and Liern [14] for repairing the constraints in order to resolve infeasibility.

The main idea is that the fuzzy membership function expresses the degree to which a particular point satisfies a given constraint. The membership function makes use

of Roodman's limits [13]. The method is stated as follows:

So as to obtain a feasible solution, let us reformulate the system

$$\mathbf{c}_i^T \mathbf{x} \geq \varepsilon_i, i = 1, \dots, k; i \neq j, \mathbf{x} \geq 0, \tag{5}$$

As the fuzzy case

$$\mathbf{c}_i^T \mathbf{x} \gtrsim \varepsilon_i, i = 1, \dots, k; i \neq j, \mathbf{x} \geq 0 \tag{6}$$

Assume that denote the fuzzy constraints by  $\tilde{C}_i, i = 1, \dots, k; i \neq k$  and their membership functions by  $\mu_{\tilde{C}_i}, i = 1, \dots, k; i \neq k$ , respectively. The construction of the membership function by Roodman's approach is based on Phase I problem (PI) associated System (5) and its dual (DPI). These problems can be stated as follows:

$$(PI) \quad \min \sum_{i=1}^k r_i$$

s.t.  $\mathbf{c}_i^T \mathbf{x} - \mathbf{I}\mathbf{s} + \mathbf{I}\mathbf{r} = \varepsilon_i, i = 1, \dots, k; i \neq j, \mathbf{x}, \mathbf{s}, \mathbf{r} \geq 0$ , where  $\mathbf{s}$  and  $\mathbf{r}$  are surplus and artificial vectors, respectively.

$$(DPI) \quad \max \sum_{i=1}^k \pi_i \varepsilon_i$$

$$s.t. \sum_{i=1}^k \pi_i c_i^j \leq 0, j = 1, \dots, n,$$

where  $c_i^j, j = 1, \dots, n$  are the elements of  $\mathbf{c}_i^T$ .

Let  $z^*$  be the optimal value of (PI) and  $\pi_i^*, i = 1, \dots, k; i \neq j$ , be the optimal solution of (DPI). Since system (5) is infeasible, then  $z^* \neq 0$ . For

$$i = 1, \dots, k; i \neq j \quad \text{set} \quad p_i = \left| \frac{z^*}{\pi_i} \right|$$

if  $\pi_i^* \neq 0$  and  $p_i = 0$  if  $\pi_i^* = 0$ . The values of  $p_i$  s are called the Rood man lower bounds. Consider the following membership function:

$$\mu_{\tilde{C}_i} = \begin{cases} 0 & \text{if } \mathbf{c}_i^T \mathbf{x} \leq \varepsilon_i - p_i \\ 1 - \frac{\varepsilon_i - \mathbf{c}_i^T \mathbf{x}}{p_i} & \text{if } \varepsilon_i - p_i \leq \mathbf{c}_i^T \mathbf{x} \leq \varepsilon_i \\ 1 & \text{if } \mathbf{c}_i^T \mathbf{x} \geq \varepsilon_i \end{cases} \tag{7}$$

for  $i = 1, \dots, k; i \neq j$ .

Define the fuzzy set of feasible solutions of system (4.1) as  $\tilde{E} = \{(x, \mu_{\tilde{E}}(x), x \in R^n)\}$  where

$$\mu_{\tilde{E}} = \min \{ \mu_{\tilde{C}_1}, \dots, \mu_{\tilde{C}_k} \}.$$

The solution with the highest degree of membership  $\tilde{E}$  is given by

$$x_{\max} = \arg \left( \max_x \min_i \{ \mu_{\tilde{C}_i} \} \right).$$

So we solve the auxiliary crisp problem:

$$(AP) \quad \min \alpha$$

$$s.t. \mathbf{c}_i^T \mathbf{x} + \alpha p_i \geq \epsilon_i, i = 1, \dots, k; i \neq j, 0 \leq \alpha \leq 1, \mathbf{x} \geq 0.$$

The next theorem proves that (AP) has feasible solution and provides a lower bound for objective function value.

**Theorem 4.1.** (AP) is feasible and its optimal value  $\alpha^*$  verifies that  $1/k \leq \alpha^* \leq 1$ , where  $k$  is the number of non-null shadow prices of (PI).

**Proof:** see [14].

In fact  $\beta^* = 1 - \alpha^*$  is the degree of feasibility.

**Example 4.3.** Consider Problem 4.2. Let DM choose the first objective as the main with the same value for the bounds of the other objectives. As mentioned in Example 4.2, the problem is infeasible.

Solving (PI) and (DPI) lead to  $z^* = 4.25, \pi_2^* = 1, \pi_3^* = 0, \pi_4^* = 1, \pi_5^* = 0.25$ . Therefore,  $p_2 = 4.25, p_3 = 0, p_4 = 4.25, p_5 = 17$ . So we must solve the following problem:

$$s.t. -2x_1 - 3x_2 - 2x_3 + 4.25\alpha \geq -6$$

$$-3x_1 + 2x_2 - 4x_3 - 0 \geq -6$$

$$x_1 + 2x_2 + 1x_3 + 4.25\alpha \geq 8$$

$$x_1 + 4x_2 + x_3 + 17\alpha \geq 18$$

$$x_1, x_2, x_3 \geq 0$$

The optimal solution is  $(x_1^*, x_2^*, x_3^*) = (0, 2.8, 0)$  with  $\alpha^* = 0.5647$ .  $\square$

In order to compare the obtained solutions by two methods described in Examples 4.2 and 4.3, we compute the satisfaction degree of the objectives with the membership functions in equations (4.3). The results are summarized in the following **Table 1**.

According to **Table 1**, the summation of satisfaction degree in our algorithm is better than the Leon and Liern method, however, the minimum degree in the Leon and Liern method is better.

### 5. Conclusion

In this paper we discussed about infeasibility, diagnosis

**Table 1. A comparison between use of the combination method and the Leon and Liern method in Examples 4.2 and 4.3.**

	The combination method	Leon and Liern method
Second objective	0.8235	0.4353
Third objective	1	1
Forth objective	0.1765	0.4353
Fifth objective	1	1
The sum of satisfaction degrees	<b>3</b>	<b>2.8706</b>

and resolution, when we used the constraint method for solving multi objective linear programming. We used elastic filter and deletion filter for finding IIS. We proposed a combination method for obtaining POSs. In the interest of repairing the constraint objectives, and for resolution of infeasibility, we proposed an algorithm, which was a combination of interactive, weighting sum and constraint method. We solved some numerical examples and compared our method with Leon and Liern method.

### REFERENCES

- [1] R. Steuer, "Multiple Criteria Optimization: Theory Computation and Application," Wiley, New York, 1986.
- [2] H. W. Kuhn and A. W. Tucker, "Nonlinear Programming," In: J. Neyman, Ed., *Proceeding of the 2nd Berkeley Symposium on Mathematical Statistics and Probability*, University of California Press, Berkeley, 1951, pp. 481-492.
- [3] Y. Y. Haimes, L. Lasdon and D. Wismer, "On a Bicriteria Formulation of the Problems of the Integrated System Identification and System Optimization," *IEEE Transaction on Systems, Man, and Cybernetics*, SMC-1, No. 3, 1971, pp. 296-297.
- [4] V. J. Bowman, "On the Relationship of the Tchebycheff Norm and the Efficient Frontier of Multi-Criteria Objectives," In: H. Thiriez and S. Zients, Eds., *Multiple-Criteria Decision Making*, Springer-Verlag, Berlin, 1976, pp. 76-86.
- [5] A. Arbel and P. Korhonen, "Using Aspiration Levels in an Interactive Interior Multi-Objective Linear Programming Algorithm," *European Journal of Operational Research*, Vol. 89, No. 1, 1996, pp. 193-201.
- [6] M. Zangiabadi, M. R. Safi and H. R. Maleki, "An Algorithm For Solving Multi Objective Programming Problems," *Logic Colloquium, ASL, European Summer Meeting*, Turin, 2004.
- [7] H. J. Zimmermann, "Fuzzy Programming and Linear Programming with Several Objective Functions," *Fuzzy Sets and Systems*, Vol. 1, No. 1, 1987, pp. 45-55. [doi:10.1016/0165-0114\(78\)90031-3](https://doi.org/10.1016/0165-0114(78)90031-3)
- [8] S. M. Guu and Y. K. Wu, "A Compromise Model for Solving Fuzzy Multi Objective Linear Programming Problems," *Journal of Chinese, Institute of Industrial Engineers*, Vol. 18, No. 5, 2001, pp. 87-93
- [9] M. Jimenez and A. Bilbao, "Pareto Optimal Solutions in Fuzzy Multi Objective Linear Programming," *Fuzzy Sets and Systems*, Vol. 160, No. 18, 2009, pp. 2714-2721. [doi:10.1016/j.fss.2008.12.005](https://doi.org/10.1016/j.fss.2008.12.005)
- [10] N. Chakravarti, "Some Results Concerning Post-Infeasibility Analysis," *European Journal of Operational Research*, Vol. 73, No. 1, 1994, pp. 139-143. [doi:10.1016/0377-2217\(94\)90152-X](https://doi.org/10.1016/0377-2217(94)90152-X)
- [11] J. W. Chinneck, "MINOS (IIS): Infeasibility Analysis Using MINOS," *Computers and Operations Research*, Vol. 21, No. 1, 1994, pp. 1-9.

- [doi:10.1016/0305-0548\(94\)90057-4](https://doi.org/10.1016/0305-0548(94)90057-4)
- [12] J. W. Chinneck, "Fast Heuristics for the Maximum Feasible Subsystem Problem," *INFORMS Journal on Computing*, Vol. 13, No. 3, 2001, pp. 210-223. [doi:10.1287/ijoc.13.3.210.12632](https://doi.org/10.1287/ijoc.13.3.210.12632)
- [13] G. M. Roodman, "Post-Infeasibility Analysis in Linear Programming," *Management Science*, Vol. 25, No. 9, 1979, pp. 916-922. [doi:10.1287/mnsc.25.9.916](https://doi.org/10.1287/mnsc.25.9.916)
- [14] T. León and V. Liern, "A Fuzzy Method to Repair Infeasibility in Linearly Constrained Problems," *Fuzzy Sets and Systems*, Vol. 122, No. 2, 2001, pp. 237-243. [doi:10.1016/S0165-0114\(00\)00010-5](https://doi.org/10.1016/S0165-0114(00)00010-5)
- [15] M. Tamiz, S. J. Mardle and D. F. Jones, "Detecting IIS in Infeasible Linear Programming Using Techniques from Goal Programming," *Computers and Operations Research*, Vol. 23, No. 2, 1996, pp. 113-119. [doi:10.1016/0305-0548\(95\)00018-H](https://doi.org/10.1016/0305-0548(95)00018-H)
- [16] J. Yang, "Infeasibility Resolution Based on Goal Programming," *Computers and Operations Research*, Vol. 35, No. 5, 2008, pp. 1483-1493. [doi:10.1016/j.cor.2006.08.006](https://doi.org/10.1016/j.cor.2006.08.006)
- [17] J. W. Chinneck, "Feasibility and Infeasibility in Optimization. Algorithms and Computational Methods," Springer, New York, 2008.
- [18] J. W. Chinneck and E. W. Dravnieks, "Locating Minimal Infeasible Constraint Sets in Linear Programs," *ORSA Journal on Computing*, Vol. 3, No. 2, 1991, pp. 157-168. [doi:10.1287/ijoc.3.2.157](https://doi.org/10.1287/ijoc.3.2.157)
- [19] J. W. Chinneck, "An Effective Polynomial-Time Heuristic for the Minimum-Cardinality IIS Set-Covering Problem," *Annals of Mathematics and Artificial Intelligence*, Vol. 17, No. 1, 1996, pp. 127-144. [doi:10.1007/BF02284627](https://doi.org/10.1007/BF02284627)
- [20] G. Brown and G. Graves, "Elastic Programming: A New Approach to Large-Scale Mixed Integer Optimization," *ORSA/TIMS Conference*, Las Vegas, 1975.