Available online at www.sciencedirect.com
ScienceDirect
Computers & Operations Research 35 (2008) 1483 – 1493
computers & operations research
www.elsevier.com/locate/cor

Infeasibility resolution based on goal programming

Jian Yang*

Rapt Inc., 625 Second Street, San Francisco, CA 94107, USA

Available online 17 October 2006

Abstract

Infeasibility resolution is an important aspect of infeasibility analysis. This paper proposes a multi-objective programming model for infeasibility resolution and develops a method based on l_∞ norm goal programming to solve the problem. Any solution is at least weakly efficient and any efficient solution is reachable by regulating the weights. For the special case of irreducibly inconsistent linear systems, any solution is guaranteed to be efficient. The method is capable of handling both linear and nonlinear/non-convex cases, as demonstrated by the numerical examples.

Keywords: Infeasibility analysis; Goal programming; IIS

1. Introduction

There are two major aspects of infeasibility analysis for mathematical programming models: (1) infeasibility diagnosis to find which constraints are inconsistent with each other; and (2) infeasibility resolution to repair the model to make it feasible. While diagnosis provides certain insights into the nature of an infeasibility, which is definitely helpful at the model building stage, the ultimate goal of infeasibility analysis is to resolve the infeasible problem. However, most published research and practice results in recent years have been focused on the diagnosis side [1–6]. Little investigation has been made in infeasibility resolution.

Techniques developed for infeasibility diagnosis might not be practically useful in automatic resolution of infeasibility. For linear models, numerous algorithms have been developed to isolate an *irreducibly inconsistent system* (IIS), which is an infeasible set of constraints whose proper subsets are all feasible. However, since an infeasible model could have multiple IISs, the repair of a single IIS may not make the whole model feasible. One tedious approach to complex infeasibilities is to repeatedly isolate and repair the individual IISs until the whole model becomes feasible. Another approach is to find the minimum cardinality IIS set cover so that the smallest number of constraints can be removed from an infeasible LP to make the remaining constraints feasible. Finding such set covers proves to be NP hard [7,8]. For nonlinear models, little attention has been paid even to the diagnosis of infeasibility due to the complexity of nonlinear programming [4], let alone the resolution of infeasibility.

Very limited literature can be found on infeasibility resolution. Charnes and Cooper [9] initiated the discussion of adjusting constraint right-hand sides to restore feasibility. Roodman [10] described how to achieve feasibility using sensitivity analysis of the phase 1 LP. McCarl [11] presented a procedure based on the big-M method to diagnose and

* Tel.: +1 415 932 2642; fax: +1 415 932 2701.
E-mail address: jimmy.yang@rapt.com.

0305-0548/$ - see front matter
doi:10.1016/j.cor.2006.08.006

repair infeasible substructures for LP. Murty et al. [5] discussed how to detect infeasibility in a linear system using the Gauss–Jordan method and how to make an infeasible system feasible using elastic programming.

However, there are two major limitations to the above methods as tools to resolve infeasibility automatically. Firstly, since all the methods are essentially based on the weighted sum of elastic variables, they are only able to find certain corner solutions. Solutions are either too sensitive to the weights or entirely insensitive to the weights. As a result, priority levels cannot be adequately represented by the weights. Secondly, no example has been provided for nonlinear models. As a matter of fact, methods based on the weighted sum would be ineffective to solve nonlinear and non-convex problems.

In this paper, the problem of infeasibility resolution is formally modeled via *multi-objective programming* (MOP), which clearly reveals the weaknesses mentioned above. To overcome these weaknesses, a method based on l_∞ norm *goal programming* (GP) is developed for infeasibility resolution. It can reach any efficient solution by regulating the weights even for nonlinear and non-convex cases. Any solution of the goal program is at least weakly efficient and can further be improved to an efficient solution by a subsequent procedure. Moreover, efficiency is guaranteed by the goal program for irreducibly inconsistent linear constraints.

The rest of the paper is organized as follows. Section 2 introduces a multi-objective programming model for infeasibility resolution and briefly reviews some relevant basic concepts. Two numerical examples, one linear and one nonlinear, are given in Section 3 to illustrate the weaknesses of the weighted sum method. Section 4 presents a method based on l_∞ norm GP and proves the efficiency properties of the solutions. The numerical examples are re-solved in Section 5 using the new method to show its effectiveness. Section 6 concludes the paper.

2. Infeasibility resolution as multi-objective programming

For a set of constraints

$$f_i(x) \leqslant 0, \quad i = 1, \ldots, m, \tag{1}$$

where $f_i : \Re^n \mapsto \Re$, define non-negative elastic or relaxation variables y_i such that

$$f_i(x) - y_i \leqslant 0, \quad i = 1, \ldots, m. \tag{2}$$

If the original constraint set (1) is feasible, then $y_i = 0$ for all $i = 1, \ldots, m$, is a solution to (2). Otherwise, some y_i must be positive. To make the constraint set feasible while keeping the relaxation as small as possible, it is natural to formulate the following MOP

$$\text{(MOP)} \quad \min \quad (y_1, \ldots, y_m) \tag{3a}$$
$$\text{s.t.} \quad f_i(x) - y_i \leqslant 0, \quad i = 1, \ldots, m, \tag{3b}$$
$$y_i \geqslant 0, \quad i = 1, \ldots, m. \tag{3c}$$

For (MOP), $(x^*, y^*) = (x_1^*, \ldots, x_n^*, y_1^*, \ldots, y_m^*)$ is an *efficient solution* (also called non-dominated, non-inferior or Pareto optimal solution) if there exists no other feasible solution $(\hat{x}, \hat{y}) = (\hat{x}_1, \ldots, \hat{x}_n, \hat{y}_1, \ldots, \hat{y}_m)$ such that $\hat{y}_i \leqslant y_i^*$ for all $i = 1, \ldots, m$ and $\hat{y} \neq y^*$. (x^*, y^*) is a *weakly efficient solution* if there exists no other feasible solution $(\hat{x}, \hat{y})$ such that $\hat{y}_i < y_i^*$ for all $i = 1, \ldots, m$. The set of all efficient solutions is called the *efficient frontier*.

One widely used method to find the efficient solutions of (MOP) is to scalarize the vector objective functions by the weighted sum (WS) as follows

$$\text{(WS)} \quad \min \quad z = \sum_{i=1}^{m} w_i y_i \tag{4a}$$
$$\text{s.t.} \quad f_i(x) - y_i \leqslant 0, \quad i = 1, \ldots, m, \tag{4b}$$
$$y_i \geqslant 0, \quad i = 1, \ldots, m, \tag{4c}$$

where $\sum_{i=1}^{m} w_i = 1$ and $w_i \geqslant 0$ for all $i = 1, \ldots, m$. The mathematical meaning of (WS) is clear: it searches for a solution with the smallest weighted average relaxation to achieve feasibility. Weight w_i represents the priority or importance of relaxation variable y_i. The larger w_i is, the less likely constraint $f_i(x) \leqslant 0$ will be relaxed. It is well known that when $w_i > 0$ for all $i = 1, \ldots, m$, any solution of (WS) is efficient [12]. A new CPLEX functionality provided by ILOG [13] deals with infeasible linear problems essentially in the same way, where the only difference is that the weights there are interpreted as the reciprocal of the preference levels.

A major weakness of the WS method is that it can never generate solutions in the concave portion of an efficient frontier. The situation is no better for linear models, where the efficient frontier is polyhedral. In theory, when the contour hyper-plane $\sum_{i=1}^{m} w_i y_i = z$ is parallel with a surface of the efficient frontier, any point on the surface could be a solution. However, even a very small numerical error in an LP solver could skew the contour hyper-plane slightly, so that it can only touch a corner of the polyhedron. In summary, solutions in both the linear case and the nonlinear/non-convex case are either too sensitive to the weights (a slight change of the weights could make a solution jump from one corner to another) or entirely insensitive to the weights (changing the weights has no effect on the solution). Priority levels cannot be well described by the weights in (WS).

3. Numerical examples

To verify what is described in Section 2, two simple numerical examples, one linear and one nonlinear, are presented.

3.1. Linear example

$$x_1 \leqslant 0, \tag{5a}$$

$$x_2 \leqslant 0, \tag{5b}$$

$$5x_1 + 6x_2 \geqslant 30. \tag{5c}$$

As shown in Fig. 1, each of the above constraints specifies a half-space in $\Re^2$. Since there is no intersection among the three half-spaces, the constraint set is infeasible. Following Section 2, a MOP can be formulated as

$$(MOP_1) \quad \min \quad (y_1, y_2, y_3) \tag{6a}$$

$$\text{s.t.} \quad x_1 - y_1 \leqslant 0, \tag{6b}$$

$$x_2 - y_2 \leqslant 0, \tag{6c}$$

$$5x_1 + 6x_2 + y_3 \geqslant 30, \tag{6d}$$

$$y_1, y_2, y_3 \geqslant 0. \tag{6e}$$

By changing the relaxation variables, the lines for constraints are moved in an attempt to make an intersection among the half-spaces. An intersection can be achieved by moving just one of the lines, which is equivalent to changing

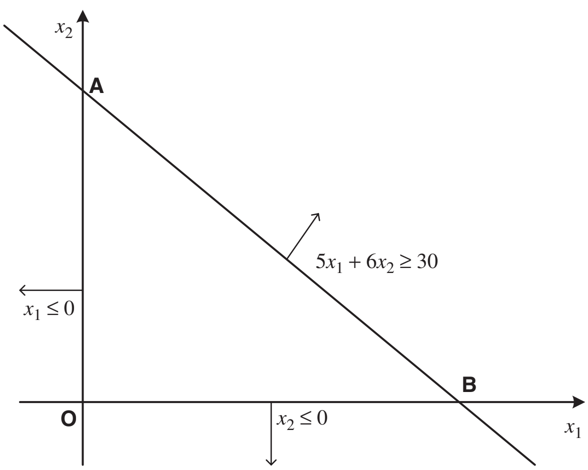

Fig. 1. Linear example.

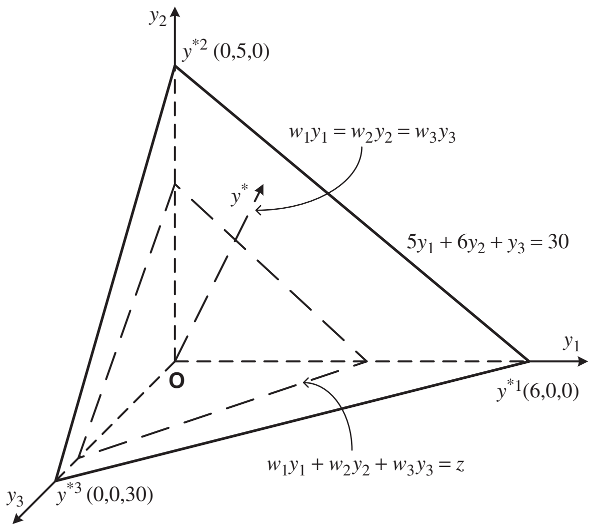

Fig. 2. Efficient frontier of linear example.

one relaxation variable while keeping the other two fixed at 0. For example, when $y_2 = 0$ and $y_3 = 0$, the minimum required value for y_1 is 6, corresponding to point B in Fig. 1. $y^{*1} = (y_1^{*1}, y_2^{*1}, y_3^{*1}) = (6, 0, 0)$ is an efficient solution in the objective space because there is no other feasible solution which is better (smaller) than y^{*1} in at least one dimension. Similarly, $y^{*2} = (y_1^{*2}, y_2^{*2}, y_3^{*2}) = (0, 5, 0)$ corresponding to point A and $y^{*3} = (y_1^{*3}, y_2^{*3}, y_3^{*3}) = (0, 0, 30)$ corresponding to point O are also efficient solutions. It can further be shown that any convex combination of the three efficient solutions, which corresponds to a point in the triangle ABO, is also efficient. Triangle ABO is actually the projection of the decision space efficient frontier onto the (x_1, x_2) space, while the objective space efficient frontier is triangle $y^{*1}y^{*2}y^{*3}$ in Fig. 2.

While the corner solutions can be interpreted as certain relaxation preferences, i.e., only one of the dimensions is allowed to be relaxed, solutions in the interior of the triangle, such as y^* in Fig. 2, might be more preferable in some situations. Unfortunately however, as indicated in Section 2, these solutions may never be reached using the following WS method

$$(WS_1)\quad \min \quad z = w_1y_1 + w_2y_2 + w_3y_3 \tag{7a}$$
$$\text{s.t.} \quad x_1 - y_1 \leqslant 0, \tag{7b}$$
$$x_2 - y_2 \leqslant 0, \tag{7c}$$
$$5x_1 + 6x_2 + y_3 \geqslant 30, \tag{7d}$$
$$y_1, y_2, y_3 \geqslant 0, \tag{7e}$$

where $w_1 + w_2 + w_3 = 1$ and $w_1, w_2, w_3 \geqslant 0$.

Given a set of weights (w_1, w_2, w_3), $z = w_1y_1 + w_2y_2 + w_3y_3$ defines a plane in the (y_1, y_2, y_3) space, shown as the dashed triangle in Fig. 2. (WS_1) pushes the plane until it just touches the efficient frontier $y^{*1}y^{*2}y^{*3}$. Different sets of weights determine different directions of the plane. However, no matter how the weights are changed to alter the direction, (WS_1) can only return the three corner solutions, as shown in the column 2 of Table 1.

3.2. Nonlinear example

Fig. 3 shows the following nonlinear example

$$x_1 + x_2 \leqslant 1, \tag{8a}$$
$$x_1x_2 \geqslant 2, \tag{8b}$$

Table 1
Solutions of linear example

(w_1, w_2, w_3)	(y_1^*, y_2^*, y_3^*)	
	Weighted sum	Goal programming
(0, 1/2, 1/2)	(6, 0, 0)	(6, 0, 0)
(1/2, 0, 1/2)	(0, 5, 0)	(0, 5, 0)
(1/2, 1/2, 0)	(0, 0, 30)	(0, 0, 30)
(1/6, 1/3, 1/2)	(6, 0, 0)	(3.6, 1.8, 1.2)
(1/6, 1/2, 1/3)	(6, 0, 0)	(4, 1.333, 2)
(1/3, 1/6, 1/2)	(0, 5, 0)	(1.698, 3.396, 1.132)
(1/3, 1/2, 1/6)	(6, 0, 0)	(2.727, 1.818, 5.455)
(1/2, 1/6, 1/3)	(0, 5, 0)	(1.224, 3.673, 1.837)
(1/2, 1/3, 1/6)	(0, 5, 0)	(1.765, 2.647, 5.294)
(3/5, 3/10, 1/10)	(0, 5, 0)	(1.304, 2.609, 7.826)
(1/3, 1/3, 1/3)	(0, 5, 0)	(2.5, 2.5, 2.5)

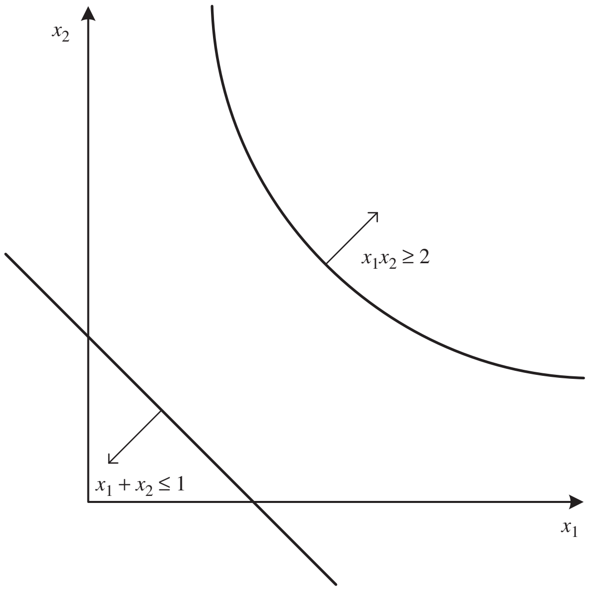

Fig. 3. Nonlinear example.

whose MOP is

$$(MOP_2)\quad \min \quad (y_1, y_2) \tag{9a}$$

$$\text{s.t.} \quad x_1 + x_2 - y_1 \leqslant 1, \tag{9b}$$

$$x_1 x_2 + y_2 \geqslant 2, \tag{9c}$$

$$y_1, y_2 \geqslant 0. \tag{9d}$$

It can be verified that the efficient frontier for (MOP_2) in the (y_1, y_2) space is

$$y_2 = 2 - \tfrac{1}{4}(y_1 + 1)^2, \tag{10a}$$

$$y_1, y_2 \geqslant 0, \tag{10b}$$

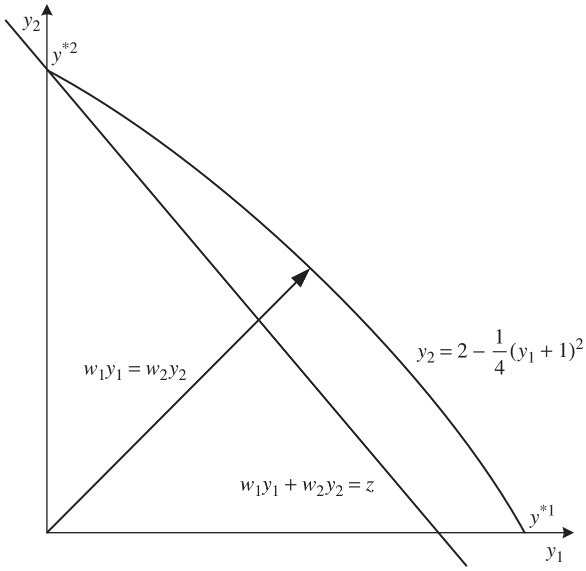

Fig. 4. Efficient frontier of nonlinear example.

Table 2
Solutions of nonlinear example

(w_1, w_2)	(y_1^*, y_2^*)	
	Weighted sum	Goal programming
(0, 1)	(1.828, 0)	(1.828, 0)
(1, 0)	(0, 1.75)	(0, 1.75)
(1/3, 2/3)	(1.828, 0)	(1.317, 0.658)
(2/3, 1/3)	(0, 1.75)	(0.657, 1.314)
(1/2, 1/2)	(0, 1.75)	(1, 1)

as shown in Fig. 4. Since the efficient frontier is concave, the following WS method can only return the two corners $y^{*1} = (y_1^{*1}, y_2^{*1}) = (2\sqrt{2} - 1, 0)$ and $y^{*2} = (y_1^{*2}, y_2^{*2}) = (0, \frac{7}{4})$, no matter how the weights are changed.

$$(WS_2) \quad \min \quad w_1 y_1 + w_2 y_2 \tag{11a}$$
$$\text{s.t.} \quad x_1 + x_2 - y_1 \leqslant 1, \tag{11b}$$
$$x_1 x_2 + y_2 \geqslant 2, \tag{11c}$$
$$y_1, y_2 \geqslant 0, \tag{11d}$$

where $w_1 + w_2 = 1$ and $w_1, w_2 \geqslant 0$. Column 2 of Table 2 shows the solutions by using different weights.

4. Goal programming for infeasibility resolution

As another popular technique for MOP, GP was first formally introduced by Charnes and Cooper [9]. Ogryczak [14] showed the relationship between GP and the reference point method. Tamiz et al. [15] gave an overview of GP for decision-making and examined its connections with other techniques of MOP, such as compromise programming and reference point method. Yang [16] developed a minimax reference point approach to facilitate goal programming in non-convex cases. Carrizosa and Romero-Morales [17] proposed a parametric problem by means of GP to combine min-sum and minimax and proved the (weak) efficiency of solutions to the parametric problem under certain conditions.

The MOP formulated for infeasibility resolution in Section 2 can be modeled as the following goal program

$$\text{(GP)} \quad \min \quad \|y - \bar{y}\| \tag{12a}$$

$$\text{s.t.} \quad f_i(x) - y_i \leqslant 0, \quad i = 1, \ldots, m, \tag{12b}$$

$$y_i \geqslant 0, \quad i = 1, \ldots, m, \tag{12c}$$

where $\bar{y} = (\bar{y}_1, \ldots, \bar{y}_m)$ is a reference point (goal) and $\|\bullet\|$ is a norm operator. An obvious reference point for infeasibility resolution is $\bar{y} = (0, \ldots, 0)$, which is also a *utopia point* or *ideal point*. Two widely used norms are weighted l_1 norm and weighted l_∞ norm:

$$\|y - \bar{y}\|_1^w = \sum_{i=1}^{m} |w_i(y_i - \bar{y}_i)|, \tag{13a}$$

$$\|y - \bar{y}\|_\infty^w = \max\{|w_i(y_i - \bar{y}_i)| : i = 1, \ldots, m\}, \tag{13b}$$

where $\sum_{i=1}^{m} w_i = 1$ and $w_i \geqslant 0$. Contours of these norms can be found in Steuer [18].

4.1. l_1 Norm goal programming

Interestingly, research on the relationship between GP and infeasibility analysis dates back to the infancy of GP. In the initial introduction of GP, Charnes and Cooper [9] also discussed its relationship to the analysis of contradictions in non-solvable LP problems as goal attainment. Tamiz et al. [13] presented an algorithm to isolate an IIS using GP. Huitzing et al. [19] presented a GP model to analyze infeasibility in test problem assembly.

However, all the above GP formulations are in one way or another based on the l_1 norm as follows:

$$(GP_1) \quad \min \quad z = \sum_{i=1}^{m} |w_i y_i| \tag{14a}$$

$$\text{s.t.} \quad f_i(x) - y_i \leqslant 0, \quad i = 1, \ldots, m, \tag{14b}$$

$$y_i \geqslant 0, \quad i = 1, \ldots, m, \tag{14c}$$

where $\sum_{i=1}^{m} w_i = 1$ and $w_i \geqslant 0$. It is obvious that (GP_1) is equivalent to (WS) under certain conditions, thus suffering the same weaknesses described in the previous sections.

4.2. l_∞ Norm goal programming

To overcome the weaknesses described earlier, the following GP_∞ is proposed, which uses the l_∞ norm for infeasibility resolution

$$(GP_\infty) \quad \min \quad z = \max\{|w_i y_i| : i = 1, \ldots, m\} \tag{15a}$$

$$\text{s.t.} \quad f_i(x) - y_i \leqslant 0, \quad i = 1, \ldots, m, \tag{15b}$$

$$y_i \geqslant 0, \quad i = 1, \ldots, m, \tag{15c}$$

where $\sum_{i=1}^{m} w_i = 1$ and $w_i \geqslant 0$. It has been shown that any efficient solution of (MOP) can be obtained by changing the weights in (GP_∞) [12], which ensures the effectiveness of using weights to represent the priority levels.

4.3. Efficiency properties of (GP_∞)

A major concern of using (GP_∞) is that its solutions are not necessarily efficient to (MOP). However, the following theorem shows that any solution to GP_∞ is at least weakly efficient.

Theorem 1 (*Weak efficiency theorem*). *If (x^*, y^*, z^*) is an optimal solution to (GP_∞), then (x^*, y^*) is a weakly efficient solution to MOP.*

Proof. (GP_∞) can be rewritten as

$$
\begin{aligned}
(GP'_\infty)\quad \min\quad & z && (16a)\\
\text{s.t.}\quad & w_i y_i \leqslant z, \quad i=1,\ldots,m, && (16b)\\
& f_i(x) - y_i \leqslant 0, \quad i=1,\ldots,m, && (16c)\\
& y_i \geqslant 0, \quad i=1,\ldots,m, && (16d)
\end{aligned}
$$

where $\sum_{i=1}^m w_i = 1$ and $w_i \geqslant 0$. The absolute sign is dropped because both w_i and y_i are non-negative. If $(x^*, y^*, z^*) = (x_1^*, \ldots, x_n^*, y_1^*, \ldots, y_m^*, z^*)$ is an optimal solution to (GP'_∞) but (x^*, y^*) is not weakly efficient to (MOP), then (MOP) must have a feasible solution $(\hat{x}, \hat{y}) = (\hat{x}_1, \ldots, \hat{x}_n, \hat{y}_1, \ldots, \hat{y}_m)$, such that $\hat{y}_i < y_i^*$ for all $i = 1, \ldots, m$. Let $\hat{z} = \sup\{w_i \hat{y}_i : i = 1, \ldots, m\}$. $(\hat{x}, \hat{y}, \hat{z})$ is a feasible solution to (GP'_∞) with smaller objective value $\hat{z} < z^*$. This is in contradiction to the assumption that (x^*, y^*, z^*) solves (GP'_∞). Therefore, (x^*, y^*) must be weakly efficient to (MOP). □

For the (GP_∞) solutions to be efficient, we need one more condition as stated in Theorem 2.

Theorem 2 (*Efficiency theorem*). *If (x^*, y^*, z^*) is an optimal solution to (GP_∞) that is unique in y^*, then (x^*, y^*) is an efficient solution to (MOP).*

Proof. Again, (GP'_∞) is used in the proof. If $(x^*, y^*, z^*) = (x_1^*, \ldots, x_n^*, y_1^*, \ldots, y_m^*, z^*)$ is an optimal solution to (GP'_∞) that is unique in y^*, by Theorem 1, (x^*, y^*) is a weakly efficient solution to (MOP). If (x^*, y^*) is not efficient to (MOP), then (MOP) must have a feasible solution $(\hat{x}, \hat{y}) = (\hat{x}_1, \ldots \hat{x}_n, \hat{y}_1, \ldots, \hat{y}_m)$, such that $\hat{y}_i \leqslant y_i^*$ for all $i = 1, \ldots, m$, and $\hat{y}_j < y_j^*$ for some $1 \leqslant j \leqslant m$. Then $(\hat{x}, \hat{y}, z^*)$ is also an optimal solution to (GP'_∞), which is in contradiction to the assumption that (x^*, y^*, z^*) is unique in y^*. Therefore, (x^*, y^*) must be efficient to (MOP). □

Note that uniqueness is the key to efficiency. The reason why (x^*, y^*) is not guaranteed to be efficient when (x^*, y^*, z^*) is not unique in y^* is that some y_i^* could have room for further reduction without affecting z^*. To get an efficient solution by squeezing out any such room, a subsequent WS problem based on the (GP_∞) result can be solved as follows:

$$
\begin{aligned}
\min\quad & z = \sum_{i=1}^m w_i y_i && (17a)\\
\text{s.t.}\quad & y_i \leqslant y_i^*, \quad i=1,\ldots,m, && (17b)\\
& f_i(x) - y_i \leqslant 0, \quad i=1,\ldots,m, && (17c)\\
& y_i \geqslant 0, \quad i=1,\ldots,m. && (17d)
\end{aligned}
$$

Fortunately, the subsequent WS procedure is not needed when $f_i(x) \leqslant 0$, $i = 1, \ldots, m$, is an irreducibly inconsistent linear system because it can be shown that any solution to (GP_∞) is unique in y^* and therefore efficient to (MOP). To prove this, we need Theorem 3 as a prerequisite.

Theorem 3 (*IIS conditions, Fan [20]*). *The linear system $Ax \leqslant b$, where A is an $m \times n$ matrix, is irreducibly inconsistent if and only if the following two conditions are simultaneously fulfilled*:

(1) *Any $m - 1$ rows of A are linearly independent.*
(2) *There exists $\lambda > 0$ such that $\lambda^{\mathrm{T}} A = 0$ and $\lambda^{\mathrm{T}} b < 0$.*

Now we can prove the following result.

Theorem 4 (*Efficiency theorem for IIS*). *Suppose that the linear system $Ax \leqslant b$, where A is an $m \times n$ matrix, is irreducibly inconsistent. Let $(x^*, y^*, z^*) = (x_1^*, \ldots, x_n^*, y_1^*, \ldots, y_m^*, z^*)^{\mathrm{T}}$ be an optimal solution to the following* *linear goal program*

$$\begin{aligned} \text{(LGP)}\ \ \min\ \ & z && \text{(18a)}\\ \text{s.t.}\ \ & Ax - Iy \leqslant b, && \text{(18b)}\\ & Wy - Jz \leqslant 0, && \text{(18c)}\\ & y_i \geqslant 0, \quad i = 1, \ldots, m, && \text{(18d)} \end{aligned}$$

where I is an $m \times m$ *identity matrix,* $J = (\overbrace{1, \ldots, 1}^{m})^{\mathrm{T}}$, $W = diag\,\{w_1, \ldots, w_m\}$, *is a diagonal matrix,* $\sum_{i=1}^{m} w_i = 1$ *and* $w_i > 0$ *for all* $i = 1, \ldots, m$. *Then* (x^*, y^*) *is an efficient solution to the following multi-objective linear program*

$$\begin{aligned} \text{(MOLP)}\ \ \min\ \ & (y_1, \ldots, y_m) && \text{(19a)}\\ \text{s.t.}\ \ & Ax - Iy \leqslant b, && \text{(19b)}\\ & y_i \geqslant 0, \quad i = 1, \ldots, m. && \text{(19c)} \end{aligned}$$

Proof. Let $\lambda = (\lambda_1, \ldots, \lambda_m)^{\mathrm{T}} \geqslant 0$ and $\mu = (\mu_1, \ldots, \mu_m)^{\mathrm{T}} \geqslant 0$ be dual variables associated with $Ax - Iy \leqslant b$ and $Wy - Jz \leqslant 0$, respectively. Note that both y and z are nonnegative but x is unrestricted. The dual problem [21] of (LGP) can be written as

$$\begin{aligned} \text{(DLGP)}\ \ \max\ \ & -\lambda^{\mathrm{T}} b && \text{(20a)}\\ \text{s.t.}\ \ & \lambda^{\mathrm{T}} A = 0, && \text{(20b)}\\ & \lambda^{\mathrm{T}} I - \mu^{\mathrm{T}} W \leqslant 0, && \text{(20c)}\\ & \mu^{\mathrm{T}} J \leqslant 1. && \text{(20d)} \end{aligned}$$

From (20b), $\lambda \in N(A^{\mathrm{T}})$, the null space of A^{T}. Because $Ax \leqslant b$ is irreducibly inconsistent, by Theorem 3 (condition 1), $\mathrm{rank}(A) = m - 1$. Therefore, $N(A^{\mathrm{T}})$ is a one-dimensional subspace, i.e., a line through the origin. By Theorem 3 (condition 2), there exists $\lambda > 0$ such that $\lambda^{\mathrm{T}} A = 0$ and $\lambda^{\mathrm{T}} b < 0$. Therefore, $\lambda^* > 0$; and from (20c), $\mu^* > 0$. Consequently, we must have that the optimal dual solution $(\lambda^*, \mu^*) = (\lambda_1^*, \ldots, \lambda_m^*, \mu_1^*, \ldots, \mu_m^*)^{\mathrm{T}} > 0$. By complementary slackness, $Ax^* - Iy^* = b$, $Wy^* - Jz^* = 0$. Therefore, (x^*, y^*, z^*) is unique in y^*. By Theorem 2, (x^*, y^*) is an efficient solution to (MOLP). □

5. Revisiting the numerical examples

To verify the effectiveness of the method developed in Section 4, the numerical examples of Section 3 are re-solved.

5.1. Linear example

$$\begin{aligned} \min\ \ & z && \text{(21a)}\\ \text{s.t.}\ \ & w_i y_i \leqslant z, \quad i = 1, 2, 3, && \text{(21b)}\\ & x_1 - y_1 \leqslant 0, && \text{(21c)}\\ & x_2 - y_2 \leqslant 0, && \text{(21d)}\\ & 5x_1 + 6x_2 + y_3 \geqslant 30, && \text{(21e)}\\ & y_1, y_2, y_3 \geqslant 0, && \text{(21f)} \end{aligned}$$

where $w_1 + w_2 + w_3 = 1$ and $w_1, w_2, w_3 \geqslant 0$.

As shown in Fig. 2, solving this problem is equivalent to moving from the origin O along an arrow in the direction determined by the weights until it reaches the efficient frontier. It is clear that any point on the efficient frontier can be reached by changing the direction of the arrow. Comparison between columns 2 and 3 of Table 1 shows the advantages of using weights in (GP_∞) to represent the priority levels.

Since (5a–c) is actually an irreducibly inconsistent system, Theorem 4 guarantees the efficiency of solutions to (GP_∞). In more general cases however, only weak efficiency can be ensured, as stated in Theorem 1. If one more constraint $-3x_1 + 5x_2 \geqslant 15$ is added to (5a–c), as shown in Fig. 5, the constraint set becomes reducibly inconsistent, or there are multiple IISs.

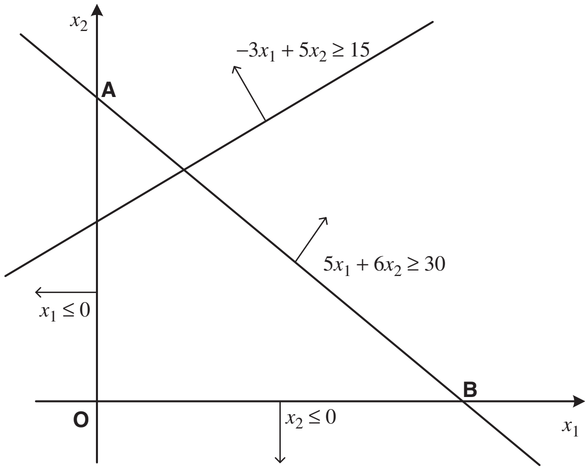

Fig. 5. Multiple IISs.

The corresponding l_∞ norm goal program is as follows:

$$\min \quad z \tag{22a}$$
$$\text{s.t.} \quad w_i y_i \leqslant z, \quad i = 1, 2, 3, 4, \tag{22b}$$
$$x_1 - y_1 \leqslant 0, \tag{22c}$$
$$x_2 - y_2 \leqslant 0, \tag{22d}$$
$$5x_1 + 6x_2 + y_3 \geqslant 30, \tag{22e}$$
$$-3x_1 + 5x_2 + y_4 \geqslant 15, \tag{22f}$$
$$y_1, y_2, y_3, y_4 \geqslant 0. \tag{22g}$$

When $w_i = 0.25$ for $i = 1, \ldots, 4$, a solution is $y^* = (y_1^*, y_2^*, y_3^*, y_4^*) = (3.235, 3.235, 3.235, 3.235)$. It is weakly efficient, but not efficient. To get an efficient solution, we can solve the following WS problem based on the previous result

$$\min \quad \sum_{i=1}^{4} w_i y_i \tag{23a}$$
$$\text{s.t.} \quad y_i \leqslant y_i^*, \quad i = 1, \ldots, 4, \tag{23b}$$
$$x_1 - y_1 \leqslant 0, \tag{23c}$$
$$x_2 - y_2 \leqslant 0, \tag{23d}$$
$$5x_1 + 6x_2 + y_3 \geqslant 30, \tag{23e}$$
$$-3x_1 + 5x_2 + y_4 \geqslant 15, \tag{23f}$$
$$y_1, y_2, y_3, y_4 \geqslant 0, \tag{23g}$$

which returns an efficient solution $\hat{y}^* = (\hat{y}_1^*, \hat{y}_2^*, \hat{y}_3^*, \hat{y}_4^*) = (1.471, 3.235, 3.235, 3.235)$.

5.2. Nonlinear example

The method developed in Section 4 can also be used to handle nonlinear and non-convex cases. The l_∞ norm goal program for (MOP_2) can be written as

$$\min \quad z \tag{24a}$$
$$\text{s.t.} \quad w_i y_i \leqslant z, \quad i = 1, 2, \tag{24b}$$
$$x_1 + x_2 - y_1 \leqslant 1, \tag{24c}$$
$$x_1 x_2 + y_2 \geqslant 2, \tag{24d}$$
$$y_1, y_2 \geqslant 0, \tag{24e}$$

where $w_1 + w_2 = 1$ and $w_1, w_2 \geqslant 0$. As shown in Fig. 4, solving this problem is equivalent to moving from the origin along an arrow in the direction determined by the weights until it reaches the efficient frontier. It is clear that any point on the efficient frontier can be reached by changing the direction of the arrow. Column 3 of Table 2 shows the solutions by using different weights. The weaknesses of the WS method presented in the previous sections are overcome.

6. Conclusions

This paper discusses infeasibility resolution as an important aspect of infeasibility analysis. A major weakness of infeasibility analysis based on a weighted sum of elastic variables is that it cannot obtain all solutions on the efficient frontier. In linear or nonlinear/non-convex problems, only corner solutions are reachable. This makes the weights less meaningful in interpreting the priority levels. To overcome the weakness, this paper develops a method based on l_∞ norm goal programming, which is capable of reaching any efficient solution by regulating the weights. Any solution of the goal program is at least weakly efficient. A weakly efficient solution can further be improved to an efficient solution using a subsequent procedure. For irreducibly inconsistent linear constraints, any solution of the goal program is guaranteed to be efficient. This method is effective in both linear and nonlinear/non-convex cases.

References

[1] Greenberg HJ. How to analyze the results of linear programs—part 3: infeasibility diagnosis. Interface 1993;23(6):120–39.
[2] Tamiz M, Mardle SJ, Jones DF. Detecting IIS in infeasible linear programmes using techniques from goal programming. Computers & Operations Research 1996;23(2):113–9.
[3] Chinneck JW. Finding a useful subset of constraints for analysis in an infeasible linear program. INFORMS Journal on Computing 1997;9(2): 164–74.
[4] Chinneck JW. Feasibility and viability. In: Gal T, Greenberg HJ, editors. Advances in sensitivity analysis and parametric programming. International series in operations research and management science, vol. 6. Boston, MA: Kluwer Academic Publishers; 1997.
[5] Murty KG, Kabadi SN, Chandrasekaran R. Infeasibility analysis for linear systems, a survey. Arabian Journal of Science and Technology 2000;25(1C):3–18.
[6] Amaldi E, Pfetsch ME, Trotter Jr. LE. On the maximum feasible subsystem problem IISs and IIS-hypergraphs. Mathematical Programming 2003;95(3):533–54.
[7] Chinneck JW. An effective polynomial-time heuristic for the minimum-cardinality IIS set-covering problem. Annals of Mathematics and Artificial Intelligence 1996;17:127–44.
[8] Huitzing HA. Using set covering with item sampling to analyze the infeasibility of linear programming test assembly models. Applied Psychological Measurement 2004;28(5):355–75.
[9] Charnes A, Cooper WW. Management models and industrial applications of linear programming. New York: Wiley; 1961.
[10] Roodman GM. Post-infeasibility analysis in linear programming. Management Science 1979;25(9):916–22.
[11] McCarl BA. Repairing misbehaving mathematical programming models: concepts and a GAMS-based approach. Interfaces 1998;28(5): 124–38.
[12] Sawaragi Y, Nakayama H, Tanino T. Theory of multiobjective optimization. London: Academic Press; 1985.
[13] ILOG. ILOG Cplex 9.0 reference manual, 2003.
[14] Ogryczak W. A goal programming model of the reference point method. Annals of Operations Research 1994;51:33–44.
[15] Tamiz M, Jones D, Romero C. Goal programming for decision making: an overview of the current state-of-the-art. European Journal of Operational Research 1998;111:569–81.
[16] Yang JB. Minimax reference point approach and its application for multiobjective optimisation. European Journal of Operational Research 2000;126:541–56.
[17] Carrizosa E, Romero-Morales D. Combining minsum and minmax: a goal programming approach. Operations Research 2001;49(1):169–74.
[18] Steuer R. Multiple criteria optimization: theory, computation, and application. New York: Wiley; 1986.
[19] Huitzing HA, Veldkamp BP, Verschoor AJ. Infeasibility in automated test assembly models: a comparison study of different methods. Measurements and Research Department Reports 2004-6, Citogroep, Arnhem, The Netherlands; 2004.
[20] Fan K. On systems of linear inequalities. Annals of Mathematical Studies 1956;38:99–156.
[21] Bertsimas D, Tsitsiklis JN. Introduction to linear optimization. Belmont, MA: Athena Scientific; 1997.