Inverse linear programming with interval coefficients

Preprint

Technical Report I/2015
Department of Econometrics
University of Economics, Prague, Czech Republic

Amin Mostafaee[a], Milan Hladík[b], Michal Černý[c]

[a] *Department of Mathematics, Islamic Azad University, North Tehran Branch, Tehran, Iran.*
[b] *Charles University, Faculty of Mathematics and Physics, Department of Applied Mathematics, Malostranské nám. 25, 11800, Prague, Czech Republic.*
[c] *University of Economics, Faculty of Informatics and Statistics, nám. W. Churchilla 4, 13067, Prague, Czech Republic.*

Abstract

The paper deals with the inverse linear programming problem over intervals. More precisely, given interval domains for the objective function coefficients and constraint coefficients of a linear program, we ask for which scenario a prescribed optimal value is attained. Using continuity of the optimal value function (under some assumptions), we propose a method based on parametric linear programming techniques. We study special cases when the interval coefficients are situated in the objective function and/or in the right-hand sides of the constraints as well as the generic case when possibly all coefficients are intervals. We also compare our method with the straightforward binary search technique. Finally, we illustrate the theory by an accompanying numerical study, called "Matrix Casino", showing some approaches to designing a matrix game with a prescribed game value.

Keywords: interval analysis, interval linear programming, inverse linear programming, parametric analysis, matrix game
2000 MSC: 65G40, 90C05

Email addresses: `a_mostafaee@iau-tnb.ac.ir` (Amin Mostafaee), `milan.hladik@matfyz.cz` (Milan Hladík), `cernym@vse.cz` (Michal Černý)

February 25, 2015

1. Introduction

Interval linear programming in general. In linear programming (LP) problems, the coefficients of the objective function, of the constraint matrix and of the right-hand sides of the constraints are usually assumed to be fixed and known input parameters. *Interval linear programming* relaxes this assumption and replaces the fixed data by known intervals of possible values. From the practical point of view, the main justification of interval linear programming is that coefficients of LP models are often not known exactly due to elicitation by inexact methods, due to subjective expert evaluations, or due to their inherent vagueness, imprecision or instability. Then it is appropriate to consider intervals of possible values of the coefficients.

When we want to use LP, it is necessary to select representatives of the intervals, for example extreme values or average values. Afterwards, we obtain an optimal solution which is optimal with respect to the chosen representatives; but it is not clear whether the solution is also optimal with respect to the real problem itself. Thus it is often appropriate to take into account all possible choices, instead of the only one determined by the selection of the representatives. Interval linear programming is *the* tool for this issue. Said roughly, interval linear programming is a possibilistic version of linear programming — it takes into account all possible scenarios within given intervals and studies what can happen "in the best and worst case".

A brief review of literature. The first papers dealing with interval LP systematically were Machost [1] and Krawczyk [2], followed by the state-of-the-art report by Beeck [3]. In the literature, much interest has been devoted to computing the bounds of optimal values, (see [4, 5, 6, 7, 8, 9, 10] among others). Determining or enclosing the set of optimal solutions of all the LP problems contained in a family of linear programming problems with interval data was considered in [11, 3, 7, 12, 13, 14, 15, 16, 2, 1]. It turned out that the fundamental results of the theory are Oettli-Prager theorem and Gerlach theorem [5]. Their generalization for the case where there is a simple dependence structure between coefficients of an interval system were derived by Hladík [17, 6].

The optimal value function. Consider an LP problem in the form

$$\min\{cx \mid Ax = b,\ x \geq 0\} \tag{1}$$

where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$. The results developed in this paper apply analogously for other LP formulations as well. We denote by

$$f(A, b, c) := \min\{cx \mid Ax = b,\ x \geq 0\}, \tag{2}$$

the optimal objective value of the linear program (1). We also define

$$f(A, b, c) = \begin{cases} +\infty, & \text{if the LP is infeasible,} \\ -\infty, & \text{if the LP is unbounded.} \end{cases}$$

Our goal. In this paper we investigate the optimal value function $f(A, b, c)$ when the entries of A, b and c are subject to independent and simultaneous perturbations in given intervals $\boldsymbol{A} = [A^L, A^U]$, $\boldsymbol{b} = [b^L, b^U]$ and $\boldsymbol{c} = [c^L, c^U]$. Thus, we have a family of LP problems

$$\min\{cx \mid Ax = b,\ x \geq 0\}, \quad A \in \boldsymbol{A},\ b \in \boldsymbol{b},\ c \in \boldsymbol{c}. \tag{3}$$

The family (3) is called *interval linear program*.

We will show that under some assumptions, the function $f(A, b, c)$ is continuous. It follows that the optimal value range

$$f(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}) = \{f(A, b, c) : A \in \boldsymbol{A}, b \in \boldsymbol{b}, c \in \boldsymbol{c}\} \tag{4}$$

is an interval and every value in the interval is attained as the optimal objective value of some problem in the family (3). The main problem of this paper is: we are devoted to finding a concrete problem, called *scenario*, in the family (3) having a prescribed optimal value. To give a precise statement, we solve the problem

$$\begin{array}{ll} \textbf{data:} & A^L, A^U \in \mathbb{R}^{m\times n};\ b^L, b^U \in \mathbb{R}^{m\times 1};\ c^L, c^U \in \mathbb{R}^{1\times n};\ \theta \in \mathbb{R} \\ \textbf{goal:} & \text{find } A_0 \in \mathbb{R}^{m\times n},\ b_0 \in \mathbb{R}^{m\times 1},\ c_0 \in \mathbb{R}^{1\times n} \\ & \text{s.t. } \min\{c_0 x \mid A_0 x = b_0,\ x \geq 0\} = \theta, \\ & \quad A^L \leq A_0 \leq A^U,\ \ b^L \leq b_0 \leq b^U,\ \ c^L \leq c_0 \leq c^U, \end{array}$$

where the relation $\leq$ between matrices/vectors is understood componentwise.

This problem enables a decision maker to set up free parameters (here, the coefficients A, b, c of a linear program) t o achieve the desired optimal value (which often measures costs or profits). Two examples of an application will be discussed in Sections 5.2 and 5.3. The former example deals with designing a matrix game with a prescribed value and the latter example deals with a problem of determining an optimal fee for playing a game.

Main results. We present an algorithm based on parametric analysis in LP [18, 19, 20, 21, 22]. This provides a new connection between parametric analysis techniques and inverse interval LP problem which is interesting from both computational and theoretical viewpoints. We compare it with a technique based on binary search. Finally, we present an application in designing matrix games. We also refer the reader to the work of Ahmed and Guam [23], which is complementary to ours.

Further remarks. Following [23], we call our approach "Inverse Interval LP" despite ambiguity of the word "inverse" used in optimization. Usually, "inverse optimization" means adjustment of cost coefficients of a given LP problem so that a known feasible solution becomes the optimal one, and the adjustment is minimal in some sense; see Ahuja and Orlin [24], Jiang et al. [25], or Zhang and Liu [26, 27]. The integer programming version of inverse optimization was studied e.g. by Schaefer [28], or Duan and Wang [29]. However, in the interval setting, new interesting questions and problems arise. Hladík [30] proposed a method to compute tolerances for the objective function and constraint coefficients such that the optimal value does not exceed prescribed bounds. Another problem is that one addressed in this paper, that is, to find (A_0, b_0, c_0) in given intervals $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c})$ attaining the prescribed optimal value θ.

2. Preliminaries

Interval matrices. An *interval* $(m \times n)$*-matrix* $\boldsymbol{A} = [A^L, A^U]$ is a family of matrices

$$\{A \in \mathbb{R}^{m\times n} \mid A^L \le A \le A^U\},$$

where $A^L, A^U \in \mathbb{R}^{m\times n}$ and $A^L \le A^U$. The matrices A^L, A^U are called *lower* and *upper bound* of $\boldsymbol{A}$, respectively. By

$$A_c := \tfrac{1}{2}(A^L + A^U), \quad A_\Delta := \tfrac{1}{2}(A^U - A^L)$$

we denote the *center matrix* and *radius matrix*, respectively. Sometimes it is advantageous to write $\boldsymbol{A} = [A^L, A^U]$ in the form

$$[A_c - A_\Delta, A_c + A_\Delta].$$

The space of all interval $(m \times n)$-matrices is denoted by $\mathbb{IR}^{m\times n}$.

An interval vector is a special case of an interval matrix; its center and radius is defined analogously.

Bounds of the optimal value range. Let $\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}$ be given. Recall that the optimal value function $f(A, b, c)$ was defined by (2) and that the optimal value range was defined by (4). The lower and upper bounds of the optimal value range are denoted, respectively, by

$$\begin{aligned} f^L &= f^L(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}) = \inf\{f(A, b, c) \mid A \in \boldsymbol{A},\ b \in \boldsymbol{b},\ c \in \boldsymbol{c}\}, \qquad (5)\\ f^U &= f^U(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}) = \sup\{f(A, b, c) \mid A \in \boldsymbol{A},\ b \in \boldsymbol{b},\ c \in \boldsymbol{c}\}. \end{aligned}$$

Observe that in general, the optimal value range need not be an interval (see Example 1). In most cases, we will study the situation when $f(A, b, c)$ is a continuous function on the compact set $\boldsymbol{A} \times \boldsymbol{b} \times \boldsymbol{c}$. Then the optimal value range is denoted by $[f^L, f^U]$.

Computation of the lower and the upper bounds of the optimal value range was addressed by many authors including Hladík [6, 7], Chinneck and Ramadan [4], Fiedler et al. [5], and Mráz [10].

Further notation. The absolute value of a vector $y \in \mathbb{R}^m$ is understood componentwise, i.e., $|y| = (|y_1|, |y_2|, \dots, |y_m|)^{\mathrm{T}}$. The all-one vector is denoted by $e = (1, 1, \dots, 1)^{\mathrm{T}}$, and e_k stands for the kth standard unit vector. The ith row and column of a matrix A are respectively denoted by A_{i*} and A_{*i}.

In LP, it is usual to speak about a decomposition of the index set $\{1, \dots, n\}$ into a set of *basic* and a set of *nonbasic* indices. The set of basic indices is denoted by B and the set of nonbasic indices is denoted by R. The symbols x_B and x_R denote the vectors of basic and nonbasic variables, respectively.

By

$$Y_m = \{y \in \mathbb{R}^m \mid |y| = e\}$$

we denote the set of all ± 1 vectors in $\mathbb{R}^m$. (For an algorithm for enumeration of elements of Y_m consult [5].)

Given a vector $y \in \mathbb{R}^m$, we denote

$$T_y = \mathrm{diag}(y_1, y_2, \dots, y_m) = \begin{pmatrix} y_1 & 0 & \dots & 0 \\ 0 & y_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & y_m \end{pmatrix}.$$

An algorithm for computation of bounds of the optimal value range. The following theorem provides an explicit formula for computing the bounds f^L, f^U of the optimal value range. Its proof can be found on p. 84 of [5].

Theorem 1. *We have*

$$f^L = \inf\{c^L x \mid A^L x \le b^U,\ A^U x \ge b^L, x \ge 0\},$$
$$f^U = \sup_{y \in Y_m} f(A_y, b_y, c^U),$$

where $A_y = A_c - T_y A_\Delta$ *and* $b_y = b_c + T_y b_\Delta$. □

Assumption. Throughout the paper we assume that f^L and f^U are finite. In particular, this means that for every $A \in \boldsymbol{A}$, $b \in \boldsymbol{b}$ and $c \in \boldsymbol{c}$, the linear programming problem $\min\{cx \mid Ax = b,\ x \ge 0\}$ is feasible and bounded.

Where the lower bound is attained. The next theorem shows that when the lower bound f^L is finite, then it is attained as the optimal value of some problem in the family (3) with a particular structure. The structure of this scenario will be useful later. For a proof see p. 89 of [5].

Theorem 2. *Let* f^L *be finite and let* x^* *be an optimal solution of the problem*

$$\min\{c^L x \mid A^L x \le b^U,\ A^U x \ge b^L, x \ge 0\}.$$

Then $f^L = f(A_c - T_y A_\Delta, b_c + T_y b_\Delta, c^L)$, *where*

$$y_i = \begin{cases} \frac{(A_c x^* - b_c)_i}{(A_\Delta x^* + b_\Delta)_i} & \text{if } (A_\Delta x^* + b_\Delta)_i > 0, \\ 1 & \text{if } (A_\Delta x^* + b_\Delta)_i = 0, \end{cases} \qquad i = 1, 2, \dots, m. \qquad \square$$

Notice that y need not belong to Y_m in general, but clearly we have

$$|y| \le e. \tag{6}$$

3. Continuity of $f(A, b, c)$

Continuity of the optimal value function f was studied in [31, 32], among others. Vranka [31] studied continuity of the optimal value function for an interval linear programming problem, under certain assumptions. Results on the continuity of the optimal value of a linear program and of related polyhedral valued multifunctions (determined by the constraints) were reviewed by Wets in [32]. He proved several results implying that $f(A, b, c)$ is a continuous function when for both the primal and dual programs and any $(A, b, c) \in (\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c})$, the optimal values and the set of optimal solutions are bounded. The proof of the next theorem can be inferred from the results of [32], but it is not explicitly stated there.

Theorem 3. *Suppose that for every $A \in \boldsymbol{A}$, $b \in \boldsymbol{b}$ and $c \in \boldsymbol{c}$ the following two conditions hold*

$$\{x \in \mathbb{R}^n \mid Ax = 0,\ x \geq 0,\ cx \leq 0\} = \{0\}, \tag{7}$$

$$\{y \in \mathbb{R}^m \mid A^T y \leq 0,\ b^T y \geq 0\} = \{0\}. \tag{8}$$

Then f^L, f^U are finite, $f(A, b, c)$ is continuous on $\boldsymbol{A} \times \boldsymbol{b} \times \boldsymbol{c}$, and the optimal solution set is bounded. □

Corollary 1. *Under the assumptions* (7)–(8)*, for every $\theta \in [f^L, f^U]$, there are $A_0 \in \boldsymbol{A}$, $b_0 \in \boldsymbol{b}$ and $c_0 \in \boldsymbol{c}$ such that $f(A_0, b_0, c_0) = \theta$.* □

Corollary 1 motivates us to seek for which scenario (A_0, b_0, c_0) the prescribed optimal value θ is attained. This problem will be addressed in Section 4. Now, we turn our attention to methods for verification of the assumptions (7)–(8).

How to verify (7)**.** The condition (7) holds true iff there is no scenario such that

$$Ax = 0,\ x \geq 0,\ cx \leq 0,\ e^{\mathrm{T}} x = 1 \tag{9}$$

is solvable. By using the results on solvability of interval systems from [5, 33], we have that (7) is satisfied iff there is no solution to the linear system

$$A^L x \leq 0,\ A^U x \geq 0,\ x \geq 0,\ c^L x \leq 0,\ e^{\mathrm{T}} x = 1. \tag{10}$$

Thus, (7) is efficiently verifiable.

How to verify (8)**.** The condition (8) holds true iff there is no scenario such that

$$A^{\mathrm{T}} y \leq 0,\ b^{\mathrm{T}} y \geq 0,\ y \neq 0 \tag{11}$$

is solvable. Again, the results on solvability of interval systems from [5, 33] imply that (8) is satisfied iff there is no solution to the linear system

$$A_c^{\mathrm{T}} y - A_\Delta^{\mathrm{T}} |y| \leq 0,\ b_c^{\mathrm{T}} y + b_\Delta^{\mathrm{T}} |y| \geq 0,\ e^{\mathrm{T}} |y| = 1. \tag{12}$$

This system is not easily solvable (we conjecture that it is NP-hard to check solvability). However, there is a natural approach to solve it. Consider a decomposition into a particular orthant. Then the description becomes linear; cf. [7]. Let $s \in Y_m$ be a sign vector corresponding to the orthant that $T_s y \geq 0$. Restricted to this orthant, (12) draws

$$(A_c^{\mathrm{T}} - A_\Delta^{\mathrm{T}} T_s) y \leq 0,\ (b_c^{\mathrm{T}} + b_\Delta^{\mathrm{T}} T_s) y \geq 0,\ (e^{\mathrm{T}} T_s) y = 1,\ T_s y \geq 0. \tag{13}$$

Now we can state that (8) is satisfied iff the system (13) is infeasible for every $s \in Y_m$. This procedure requires solving 2^m linear programs, which is practicable only for mild dimensions.

Remark 1. The general negative result does not rule out a possibility that in particular special cases, verification of the condition (8) might be easier. We will show an example in Proposition 2.

Example 1. Consider the interval linear program

$$\min -x_1 \text{ subject to } \lambda x_1 + x_2 = 6,\ 2x_1 + x_2 + x_3 = 6,\ x_1, x_2, x_3 \geq 0,$$

where $\lambda \in [1,2]$. It is easy to see that for $\lambda \in [1,2)$, the optimal value $f(\lambda) = 0$, but for $\lambda = 2$ we get $f(\lambda) = -3$. Thus, the optimal value range $f([1,2]) = \{-3, 0\}$ is discontinuous in spite of the finite extremal optimal values $f^L = -3$ and $f^U = 0$.

Applying our method, the condition (7) turns out to be satisfied. To check (8), set up the system (12)

$$\begin{aligned} 1.5y_1 + 2y_2 - 0.5|y_1| &\leq 0, \\ y_1 + y_2 &\leq 0, \\ y_2 &\leq 0, \\ 6y_1 + 6y_2 &\geq 0, \\ |y_1| + |y_2| &= 1. \end{aligned}$$

Since $y_2 \leq 0$, we do not have to inspect the orthants with $s = (1,1)$ and $s = (-1,1)$. For $s = (-1,-1)$, the system (13) is infeasible, but for $s = (1,-1)$, the system (13) has a solution $y = \frac{1}{2}(1,-1)^{\mathrm{T}}$.

Remark 2. If the LP problem (1) would be in the form of

$$\min\{cx \mid Ax \geq b,\ x \geq 0\}, \tag{14}$$

then the conditions (7)–(8) read

$$\{x \in \mathbb{R}^n \mid Ax \geq 0,\ x \geq 0,\ cx \leq 0\} = \{0\}, \tag{15}$$

$$\{y \in \mathbb{R}^m \mid A^{\mathrm{T}}y \leq 0,\ y \geq 0,\ b^{\mathrm{T}}y \geq 0\} = \{0\}. \tag{16}$$

Now, (15) is satisfied iff there is no solution to

$$A^U x \geq 0,\ x \geq 0,\ c^L x \leq 0,\ e^{\mathrm{T}}x = 1,$$

and (16) is satisfied iff there is no solution to

$$(A^L)^{\mathrm{T}}y \leq 0,\ y \geq 0,\ (b^U)^{\mathrm{T}}y \geq 0,\ e^{\mathrm{T}}y = 1.$$

Hence both assumptions are easy to verify. Indeed, the form (14) of an LP is known to have nice properties from the interval analysis viewpoint; see [7].

4. Inverse interval LP problem

Assumptions and goal. First we summarize our assumptions:

(i) assumptions (7)–(8) are satisfied;

(ii) the values f^L and f^U are known and finite;

(iii) a number $\theta \in [f^L, f^U]$, called *demand*, is given;

(iv) the minimizers $(A_L^*, b_L^*, c_L^*) \in (\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c})$ such that $f(A_L^*, b_L^*, c_L^*) = f^L$ are known;

(v) the maximizers $(A_U^*, b_U^*, c_U^*) \in (\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c})$ such that $f(A_U^*, b_U^*, c_U^*) = f^U$ are known.

The objective is to find $A_0 \in \boldsymbol{A}$, $b_0 \in \boldsymbol{b}$, $c_0 \in \boldsymbol{c}$ such that $\theta = f(A_0, b_0, c_0)$.

In the next (sub)sections we will study several special cases of the inverse interval LP problem — a problem with interval cost coefficients, a problem with interval right-hand sides, a problem with both interval cost coefficients and right-hand sides. Then we combine the methods to derive an algorithm for the general case where any coefficient of a linear program may be interval.

Parametric programming. The basic (rough) idea of our approach is to start with the scenario A_L^*, b_L^*, c_L^* corresponding to the best case optimal value f^L and then "shift" the data towards the worst case scenario A_U^*, b_U^*, c_U^*. By the continuity of f, the value θ is attained at some moment.

To achieve our goal, we apply parametric analysis techniques for linear programming. We will build on the classical basis stability approach [18, 19, 20, 21] despite the recent progress in alternative ways [34, 35, 36, 37]. Since we need to use parametric programming with large perturbations, and not just a local sensitivity analysis, the classical approach is the right tool for our problem.

Recall that the index sets $B, R \subseteq \{1, \ldots, n\}$ denote the basic and nonbasic variables, respectively. By A_B we denote the restriction of A to the basic columns, and similarly for c_B. Analogous notation is used for the nonbasic indices.

A basis B is optimal if the following two conditions simultaneously hold:

1. feasibility condition: $A_B^{-1} b \geq 0$,
2. optimality condition: $c_R - c_B A_B^{-1} A_R \geq 0$.

The first condition ensures that the vector x^* defined as $x_B^* := A_B^{-1} b$, $x_R^* = 0$ is feasible and the second one implies that x^* is optimal. In parametric programming, these conditions are utilized to characterize stability regions for maximal admissible perturbations of the entries of A, b and c preserving the optimality of B.

Suppose that A, b and c are not constant, but depend on a parameter λ, so we denote them by $A(\lambda)$, $b(\lambda)$ and $c(\lambda)$. The region of admissible values of λ, for which B remains optimal, is then described by the inequality system

$$A(\lambda)_B^{-1} b(\lambda) \geq 0, \tag{17}$$

$$c(\lambda)_R - c(\lambda)_B A(\lambda)_B^{-1} A(\lambda)_R \geq 0 \tag{18}$$

4.1. Case I: Only cost coefficients are interval

First, we start with the case $A = A^L = A^U$ and $b = b^L = b^U$. Only the objective vector c can vary in $\boldsymbol{c} = [c^L, c^U]$.

In Case I, and also in the subsequent Case II, just the finiteness of both f^L and f^U implies continuity of the optimal value function [31]. So the conditions (7)–(8) need not be verified.

Theorem 1 tells us that

$$f^L(A, b, \boldsymbol{c}) = f(A, b, c^L), \quad f^U(A, b, \boldsymbol{c}) = f(A, b, c^U).$$

From Assumption (i) we also know that $f(A, b, c^L)$ and $f(A, b, c^U)$ are finite. We try to find $\lambda \in [0, 1]$ such that

$$\theta = f(A, b, (1 - \lambda)c^L + \lambda c^U) = f(A, b, c^L + \lambda(c^U - c^L)).$$

So we are perturbing c along the direction $c^U - c^L = 2c_\Delta$. Notice that in Case I, the system (17)–(18) reads

$$A_B^{-1} b \geq 0, \tag{19}$$

$$c_R^L + \lambda(c_R^U - c_R^L) - (c_B^L + \lambda(c_B^U - c_B^L))A_B^{-1} A_R \geq 0, \tag{20}$$

and it is a linear system in λ.

Let $LP(A, b, c)$ denote the linear program $\min\{c^{\mathrm{T}} x : Ax = b, x \geq 0\}$. Defining

$$\theta(\lambda) = f(A, b, (1 - \lambda)c^L + \lambda c^U),$$

we have

$$\theta(0) = f(A, b, c^L) = c^L x^1,$$
$$\theta(1) = f(A, b, c^U) = c^U x^*$$

for some $x^1, x^* \in X := \{x \mid Ax = b,\ x \geq 0\}$.

Denote by B_1 an optimal basis corresponding to x^1, and define $\lambda_1 = \max\{\lambda \in [0, 1] : B_1$ is an optimal basis for $LP(A, b, (1 - \lambda)c^L + \lambda c^U)\}$. The value λ_1 is easily calculated from (19)–(20). For $\lambda \in (\lambda_1, 1]$, the basis B_1 is no longer optimal. (Note that x^1 still can be an optimal solution corresponding to another optimal basis.) For $\lambda := \lambda_1$, a new adjacent basis B_2 becomes optimal. (B_2 can be easily found by the simplex method.) With B_2 we establish a new basis stability interval $[\lambda_1, \lambda_2]$. The process is repeated until we reach an LP the objective value of which is $\geq \theta$.

It is easy to see that $\theta(\lambda)$ is a continuous and piecewise linear function of λ. Moreover, it is linear on each stability region $[\lambda_{k-1}, \lambda_k]$ with the basis B_k since

$$c(\lambda)x^k = c^L x^k + \lambda(c^U - c^L)x^k,$$

where x^k is the optimal solution corresponding to B_k.

Clearly, $\theta(\lambda_{k-1}) \le \theta \le \theta(\lambda_k)$ for some k. Setting

$$\mu = \frac{\theta - \theta(\lambda_{k-1})}{\theta(\lambda_k) - \theta(\lambda_{k-1})},$$

we have

$$\begin{aligned}
\theta &= (1-\mu)\theta(\lambda_{k-1}) + \mu\theta(\lambda_k) \\
&= ((1-\mu)((1-\lambda_{k-1})c^L + \lambda_{k-1}c^U) + \mu((1-\lambda_k)c^L + \lambda_k c^U))x^k \\
&= (((1-\mu)(1-\lambda_{k-1}) + \mu(1-\lambda_k))c^L + ((1-\mu)\lambda_{k-1} + \mu\lambda_k)c^U)x^k.
\end{aligned}$$

This in turn implies that

$$\theta = f(A, b, \alpha_1 c^L + \alpha_2 c^U),$$

where

$$\begin{aligned}
\alpha_1 &= (1-\mu)(1-\lambda_{k-1}) + \mu(1-\lambda_k), \\
\alpha_2 &= (1-\mu)\lambda_{k-1} + \mu\lambda_k.
\end{aligned}$$

Observe that $0 \le \mu \le 1$, $\alpha_1 + \alpha_2 = 1$, $\alpha_1 \ge 0$ and $\alpha_2 \ge 0$. The scenario $c \in \boldsymbol{c}$, for which the optimal value θ is attained, is $c := \alpha_1 c^L + \alpha_2 c^U$.

Example 2. Consider the following interval LP problem:

$$\begin{array}{lrcrcl}
\text{min} & [-1,5]x_1 & + & [-3,0]x_2 & & \\
\text{subject to} & x_1 & + & x_2 & \le & 6 \\
 & -x_1 & + & 2x_2 & \le & 6 \\
 & x_1 & & & \ge & 0 \\
 & & & x_2 & \ge & 0.
\end{array}$$

The lower bound and the upper bound of the cost vector are, respectively,

$$c^L = (-1, -3), \quad c^U = (5, 0).$$

Now we have

$$f^L = f(A, b, c^L) = -14, \quad f^U = f(A, b, c^U) = 0$$

and

- the argmin of the LP problem $\min\{c^L x : Ax \le b,\ x \ge 0\}$ is $x^1 = (2, 4)^{\mathrm{T}}$,

- the argmin of the LP problem $\min\{c^U x : Ax \le b,\, x \ge 0\}$ is $x^* = (0,0)^{\mathrm{T}}$.

Let the demand $\theta = -2$ be given. We want to find $c_0 \in [c^L, c^U]$ such that $f(A,b,c_0) = -2$. We try to obtain a suitable $\lambda \in [0,1]$ for which

$$\theta(\lambda) = f(A, b, (-1+6\lambda, -3+3\lambda)) = -2.$$

The optimal solution corresponding to $\theta(\lambda)$ with $\lambda = 0$ is $x^1 = (2,4)^{\mathrm{T}}$. From (20) we calculate that the current solution x^1 remains optimal for each $\lambda \in [0, \lambda_1] = [0, \frac{1}{3}]$. Since $\theta(\lambda_1) = -6 < -2$, we proceed to the neighboring stability interval. In λ_1 we move to the adjacent basic solution $x^2 = (0,3)^{\mathrm{T}}$. From (20) we get that x^2 is optimal for each $\lambda \in [\lambda_1, \lambda_2] = [\frac{1}{3}, 1]$. Thus, the desired value of λ lies within this interval. On $[\lambda_1, 1]$, the optimal value is given by

$$\theta(\lambda) = c(\lambda)x^2 = -9 + 9\lambda,$$

so the goal -2 is attained at $\lambda = \frac{7}{9}$. The appropriate cost coefficient vector is

$$c_0 = \tfrac{2}{9}(-1,-3) + \tfrac{7}{9}(5,0) = (\tfrac{11}{3}, -\tfrac{2}{3}),$$

and the corresponding scenario reads

$$\begin{array}{rclrcrcl}
-2 & = & \min & \frac{11}{3}x_1 & - & \frac{2}{3}x_2 & & \\
 & & \text{subject to} & x_1 & + & x_2 & \le & 6 \\
 & & & -x_1 & + & 2x_2 & \le & 6 \\
 & & & x_1 & & & \ge & 0 \\
 & & & & & x_2 & \ge & 0.
\end{array}$$

4.2. Case II: Only right-hand sides are interval

Now we inspect the case $A = A^L = A^U$ and $c = c^L = c^U$. Only the right-hand side vector b can be perturbed over $\boldsymbol{b} = [b^L, b^U]$.

Suppose also that we have

- an optimal basis B_1 of the problem $\min\{cx \mid Ax = b_L^*, x \ge 0\}$ $(= f^L)$,
- an optimal basis B_* of the problem $\min\{cx \mid Ax = b_U^*, x \ge 0\}$ $(= f^U)$.

Given $\theta \in [f^L, f^U]$, we aim at finding $\lambda \in [0,1]$ such that

$$\theta = \theta(\lambda),$$

where

$$\theta(\lambda) = f(A, (1-\lambda)b_L^* + \lambda b_U^*, c) = f(A, b_L^* + \lambda(b_U^* - b_L^*), c).$$

Thus, the right hand side is perturbed along the vector $b_U^* - b_L^*$. Notice that the system (17)–(18) reads

$$\begin{aligned}
A_B^{-1}b_L^* + \lambda A_B^{-1}(b_U^* - b_L^*) &\ge 0,\\
c_R - c_B A_B^{-1} A_R &\ge 0,
\end{aligned}$$

and it is a linear system in λ. Thus, we easily determine the largest interval $[0, \lambda_1]$ for which B_1 remains optimal. If $\theta(\lambda_1) < \theta$, we proceed to the adjacent basis B_2 and the corresponding stability interval $[\lambda_1, \lambda_2]$. We repeat the process until we arrive at a basis B_k and the corresponding stability interval $[\lambda_{k-1}, \lambda_k]$ such that

$$\theta(\lambda_{k-1}) < \theta \leq \theta(\lambda_k).$$

On this interval, the optimal solution is

$$x^k_{B_k}(\lambda) = A^{-1}_{B_k}\left(b^*_L + \lambda(b^*_U - b^*_L)\right)$$

and the optimal value is

$$\theta(\lambda) = cx^k(\lambda) = c_{B_k}A^{-1}_{B_k}\left(b^*_L + \lambda(b^*_U - b^*_L)\right).$$

Hence, θ is attained at

$$\lambda = \frac{\theta - c_{B_k}A^{-1}_{B_k}b^*_L}{c_{B_k}A^{-1}_{B_k}(b^*_U - b^*_L)}.$$

4.3. Case III: Both right-hand sides and cost coefficients are interval

In this section we inspect the case where $A = A^L = A^U$, and both the right-hand side vector $\boldsymbol{b}$ and the objective vector $\boldsymbol{c}$ are interval-valued. A value $\theta \in [f^L, f^U]$ is given and we want to find $b_0 \in \boldsymbol{b}$ and $c_0 \in \boldsymbol{c}$ such that $f(A, b_0, c_0) = \theta$.

Let

$$f^L(A, \boldsymbol{b}, \boldsymbol{c}) = f(A, b^*_L, c^L), \quad f^U(A, \boldsymbol{b}, \boldsymbol{c}) = f(A, b^*_U, c^U).$$

The optimal value function with respect to λ reads

$$\begin{aligned}\theta(\lambda) &= f(A, (1-\lambda)b^*_L + \lambda b^*_U, (1-\lambda)c_L + \lambda c_U)\\ &= f(A, b^*_L + \lambda(b^*_U - b^*_L), c^L + \lambda(c^U - c^L)).\end{aligned}$$

The vector c^L is perturbed along the direction $c^U - c^L$ and simultaneously the vector b^*_L is perturbed along the direction $b^*_U - b^*_L$. In this case, the conditions (17)–(18) reduce to

$$A^{-1}_B b^*_L + \lambda A^{-1}_B(b^*_U - b^*_L) \geq 0, \tag{21}$$

$$c^L_R + \lambda(c^U_R - c^L_R) - (c^L_B + \lambda(c^U_B - c^L_B))A^{-1}_B A_R \geq 0. \tag{22}$$

Again, this is a linear system of inequalities with respect to λ, so we can proceed in the same manner as for Cases I and II. On each stability interval, the optimal solution corresponding to a basis B_k is

$$x^k_{B_k}(\lambda) = A^{-1}_{B_k}\left(b^*_L + \lambda(b^*_U - b^*_L)\right)$$

and the optimal value is

$$\theta(\lambda) = c(\lambda)x^k(\lambda) = \left(c^L_{B_k} + \lambda(c^U_{B_k} - c^L_{B_k})\right)A^{-1}_{B_k}\left(b^*_L + \lambda(b^*_U - b^*_L)\right).$$

Herein, $\theta(\lambda)$ is a quadratic function, but determining for which value λ the value θ is attained is still an easy task.

Example 3. Consider the following problem:

$$\begin{array}{llllllll}
\text{min} & [-1,5]x_1 & + & [-3,0]x_2 & & & & \\
\text{subject to} & x_1 & + & x_2 & + & x_3 & & = & [1,6] \\
& -x_1 & + & 2x_2 & & & + & x_4 & = & [6,10] \\
& & & & & & & x & \geq & 0.
\end{array}$$

We have

$$f^L(A,\boldsymbol{b},\boldsymbol{c}) = f(A,b^U,c^L) = -\tfrac{50}{3}, \quad f^U(A,\boldsymbol{b},\boldsymbol{c}) = f(A,b^L,c^U) = 0.$$

That is, in this example, the lowest optimal value is attained for $b_L^* := b^U$ and the greatest one for $b_U^* := b^L$.

Let the demand $\theta = -7$ be given, so we are requested to find $b_0 \in \boldsymbol{b}$ and $c_0 \in \boldsymbol{c}$ such that $f(A, b_0, c_0) = -7$. The right-hand side vector is perturbed along the direction $b_U^* - b_L^* = (-5,-4)^{\mathrm{T}}$ and the vector of cost coefficients is perturbed along the direction $c_U - c_L = (6,3)$. At $\lambda = 0$, the optimal basis is $B_1 = \{1,2\}$. From (21)–(22) we compute the stability interval, under which B_1 remains optimal, to be $[0, \lambda_1] = [0, \frac{1}{3}]$.

At λ_1, we switch to the adjacent basis $B_2 = \{2,4\}$. The stability interval is $[\lambda_1, \lambda_2] = [\frac{1}{3}, 1]$ and the optimal value function on this interval draws

$$\theta(\lambda) = c(\lambda)x^2(\lambda) = -18 + 33\lambda - 15\lambda^2.$$

The value of $\theta = -7$ is attained at $\lambda \approx 0.41$. That is, the scenario in question is

$$\begin{array}{llllllll}
-7 & = & \text{min} & 1.46x_1 & - & 1.77x_2 & & \\
& & \text{subject to} & x_1 & + & x_2 & \leq & 3.95 \\
& & & -x_1 & + & 2x_2 & \leq & 8.36 \\
& & & x_1 & & & \geq & 0 \\
& & & & & x_2 & \geq & 0.
\end{array}$$

4.4. The general case — all coefficients are interval

Herein, we present a general method for determination of a scenario (A_0, b_0, c_0) in $(\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c})$ with the prescribed optimal value $\theta \in [f^L, f^U]$. First, we show that the desired scenario may be searched for in a specific subset of $\boldsymbol{A} \times \boldsymbol{b} \times \boldsymbol{c}$.

Proposition 1. *Every $\theta \in [f^L, f^U]$ is attained for an LP in the form*

$$c = (1-\lambda)c^L + \lambda c^U, \quad A = A_c - T_y A_\Delta, \quad b = b_c + T_y b_\Delta, \tag{23}$$

where $\lambda \in [0,1]$ and $y \in [-1,1]^m$.

Proof. By Theorems 1 and 2, f^L is attained for

$$c = c^L, \quad A = A_c - T_{y^1} A_\Delta, \quad b = b_c + T_{y^1} b_\Delta,$$

where $y^1 \in [-1,1]^m$, and f^U is attained for

$$c = c^U, \quad A = A_c - T_{y^2} A_\Delta, \quad b = b_c + T_{y^2} b_\Delta,$$

where $y^2 \in \{\pm 1\}^m \subset [-1,1]^m$. Due to the continuity of $f(A,b,c)$, the value of θ is attained for a convex combination of these scenarios, which can be expressed as (23). $\square$

Proposition 1 (and its proof) will help us design a parametric method for finding the desired scenario. Denote by A^1, b^1 and c^1 the scenario for which f^L is attained, and by A^2, b^2 and c^2 the scenario corresponding to f^U.

It is tempting to consider the convex combination

$$A = (1-\lambda)A^1 + \lambda A^2, \quad b = (1-\lambda)b^1 + \lambda b^2, \quad c = (1-\lambda)c^1 + \lambda c^2,$$

of the above two scenarios, where $\lambda \in [0,1]$ is a parameter. However, dealing with such parametric programs is difficult in general despite the fact that there is only one parameter. A tractable case is for example when $\operatorname{rank}((A^1 \mid b^1) - (A^2 \mid b^2)) \leq 1$, that is, the parameter appears only in one constraint or in the coefficients corresponding to one variable.

We propose the following method, which is based on a movement from $(A^1 \mid b^1)$ to $(A^2 \mid b^2)$ sequentially row by row, and not at once. Thus, the parameter is situated in one constraint only, which is easier to handle; see also Grygarová [38].

The method. First, consider the parametric program with

$$A = A^1, \quad b = b^1 \quad c = (1-\lambda)c^1 + \lambda c^2.$$

We solve this problem in the same manner as in Section 4.1. If θ is attained for some value of the parameter $\lambda \in [0,1]$, then we are done. Notice that $f(A^1, b^1, (1-\lambda)c^1 + \lambda c^2)$ is an increasing function of λ. Therefore, when $f(A^1, b^1, c^2) < \theta$, solving this parametric program is not necessary.

Otherwise, we sequentially for $k = 1, \dots, m$ run the following two procedures until θ is achieved. In the first stage, consider the parametric program with

$$\begin{aligned}
A_{i*} &= A^2_{i*},\ b_{i*} = b^2_{i*}, \quad i = 1, \dots, k-1, \\
b_{k*} &= (1-\lambda)b^1_{k*} + \lambda b^2_{k*}, \\
A_{i*} &= A^1_{i*}, \quad i = k, \dots, m, \\
b_{i*} &= b^1_{i*}, \quad i = k+1, \dots, m, \\
c &= c^2.
\end{aligned}$$

The parameter λ appears only in the right-hand side in the kth entry. Thus, it is easily solved by the lines of Section 4.2. In the second stage, consider the parametric program with

$$\begin{aligned}
A_{i*} &= A^2_{i*}, \quad i = 1, \dots, k-1, && (24) \\
b_{i*} &= b^2_{i*}, \quad i = 1, \dots, k, \\
A_{k*} &= (1-\lambda)A^1_{k*} + \lambda A^2_{k*}, && (25) \\
A_{i*} &= A^1_{i*}, \quad i = k+1, \dots, m, && (26) \\
b_{i*} &= b^1_{i*}, \quad i = k+1, \dots, m, \\
c &= c^2.
\end{aligned}$$

Now, the parameter λ appears only in the kth constraint in the left hand side. The stability region of λ for a basis B is determined according to the feasibility and optimality criteria (17)–(18), which take the form

$$\begin{aligned} A(\lambda)_B^{-1}b &\geq 0,\\ c_R - c_B A(\lambda)_B^{-1} A(\lambda)_R &\geq 0. \end{aligned}$$

We have to explicitly compute the inverse of the parametric matrix $A(\lambda)_B$. Since the matrix has the form of $A(\lambda)_B = M + \lambda e_k d^{\mathrm{T}}$ for some vector $d \in \mathbb{R}^m$ and some matrix $M \in \mathbb{R}^{m\times m}$, we can employ the well-known Sherman–Morrison formula to express the inverse matrix. Then the first criterion reads

$$A(\lambda)_B^{-1}b = (M + \lambda e_k d^{\mathrm{T}})^{-1}b = M^{-1}b - \frac{\lambda}{1+\lambda d^{\mathrm{T}} M^{-1}_{*k}} M^{-1}_{*k} d^{\mathrm{T}} M^{-1} b \geq 0,$$

The denominator should be positive, giving raise to the restriction

$$\lambda < -\frac{1}{d^{\mathrm{T}} M^{-1}_{*k}} \tag{27}$$

provided $d^{\mathrm{T}} M^{-1}_{*k} < 0$. Next, the inequalities can be rewritten as

$$M^{-1}b + \lambda(M^{-1}b d^{\mathrm{T}} M^{-1}_{*k} - M^{-1}_{*k} d^{\mathrm{T}} M^{-1} b) \geq 0.$$

To obtain the stability interval for λ is a trivial task now.

Similarly we proceed for the optimality criterion:

$$\begin{aligned} &c_R - c_B A(\lambda)_B^{-1} A(\lambda)_R\\ &\quad = c_R - c_B (M + \lambda e_k d^{\mathrm{T}})^{-1} A(\lambda)_R\\ &\quad = c_R - c_B \left(M^{-1} - \frac{\lambda}{1+\lambda d^{\mathrm{T}} M^{-1}_{*k}} M^{-1}_{*k} d^{\mathrm{T}} M^{-1} \right) A(\lambda)_R \geq 0. \end{aligned}$$

This can be simplified to a system of inequalities

$$\begin{aligned} &c_R - c_B M^{-1} A(\lambda)_R\\ &\quad + \lambda \left((c_R - c_B M^{-1} A(\lambda)_R) d^{\mathrm{T}} M^{-1}_{*k} - c_B M^{-1}_{*k} d^{\mathrm{T}} M^{-1} A(\lambda)_R \right) \geq 0. \end{aligned}$$

Since $A(\lambda)_R$ depends linearly on λ, the system contains quadratic functions with respect to λ, and the stability interval for B can be straightforwardly determined.

Singularity. If the stability interval for λ is closed, we can simply move to the neighboring stability interval by a basis change. If the stability interval is semi-closed due to the restriction (27), then it has the form $[\lambda_1, \lambda_2)$, and the basis $A(\lambda)_B$ is singular for $\lambda := \lambda_2$. However, on $[\lambda_1, \lambda_2)$, the optimal value reads

$$c_B A(\lambda)_B^{-1} b = c_B M^{-1} b - \frac{\lambda}{1+\lambda d^{\mathrm{T}} M^{-1}_{*k}} c_B M^{-1}_{*k} d^{\mathrm{T}} M^{-1} b.$$

The denominator tends to zero as $\lambda \to \lambda_2$, but the optimal value function $f(A, b, c)$ is continuous, which implies that the second term vanishes and the optimal value is constant on $[\lambda_1, \lambda_2)$. Due to continuity of $f(A, b, c)$ again, we have that it is constant on $[\lambda_1, \lambda_2]$, and we can move to the neighboring stability interval based on a basis corresponding to $\lambda := \lambda_2$.

Summary. Our method requires solving up to $2m + 1$ one-parametric LP problems. Notice that the optimal value function needn't be non-decreasing, but the intermediate value theorem ensures that each value $\theta \in [f^L, f^U]$ is eventually reached.

Example 4. Consider the following interval LP problem:

$$
\begin{array}{l}
\min \\
\quad [-2,-1]x_1 + [-1,1]x_2 + [-1,2]x_3 + [-2,-1]x_4 + [\ \ 2,\ \ 4]x_5 + [\ \ 0,0]x_6 \\
\text{subject to} \\
\quad [\ \ 2,\ \ 3]x_1 + [\ \ 1,4]x_2 + [\ \ 2,7]x_3 + [\ \ 1,\ \ 1]x_4 + [-2,-2]x_5 + [\ \ 0,1]x_6 = [6,\ \ 8] \\
\quad [-1,\ \ 1]x_1 + [\ \ 1,3]x_2 + [\ \ 1,3]x_3 + [\ \ 2,\ \ 3]x_4 + [-2,\ \ 1]x_5 + [-1,0]x_6 = [6,10] \\
\quad [\ \ 1,\ \ 1]x_1 + [\ \ 1,2]x_2 + [\ \ 2,3]x_3 + [-1,\ \ 1]x_4 + [\ \ 1,\ \ 1]x_5 + [\ \ 1,1]x_6 = [8,12] \\
\hfill x_1, x_2, x_3, x_4, x_5, x_6 \geq 0.
\end{array}
$$

By using the formulas provided in Theorem 1, the lower bound and the upper bound of the optimal value range are $f^L = -112$ and $f^U = 128$, respectively. Let $\theta = 50 \in [-112, 128]$ be given. We want to find a scenario (A_0, b_0, c_0) such that $f(A_0, b_0, c_0) = 50$. We easily observe that $f(A^1, b^1, c^2) = 20$ and $f(A^2, b^2, c^2) = 128$, where

$$
\begin{aligned}
A^1 &= A_c - T_{y^1} A_\Delta, \\
b^1 &= b_c + T_{y^1} b_\Delta = (8, 10, 12)^{\mathrm{T}}, \\
A^2 &= A_c - T_{y^2} A_\Delta, \\
b^2 &= b_c + T_{y^2} b_\Delta = (6, 10, 12)^{\mathrm{T}}, \\
y^1 &= (1, 1, 1), \\
y^2 &= (-1, 1, 1), \\
c^2 &= (-1, 1, 2, -1, 4, 0).
\end{aligned}
$$

Note that the values $f(A^1, b^1, c^2) = 20$ and $f(A^2, b^2, c^2) = 128$ are already available when computing the upper bound by Theorem 2.

Since $f(A^1, b^1, c^2) = 20 < 50 = \theta$, the value θ is not attained by solving the parametric problem $f(A^1, b^1, (1 - \lambda)c^1 + \lambda c^2)$. We proceed to the first stage of the iterative process. By setting $k = 1$, we have

$$
\begin{array}{lrcrcrcrcrcrcl}
\min & -x_1 & + & x_2 & + & 2x_3 & - & x_4 & + & 4x_5 & & & & \\
\text{subject to} & 2x_1 & + & x_2 & + & 2x_3 & + & x_4 & - & 2x_5 & & & = & 8 - 2\lambda \\
& -x_1 & + & x_2 & + & x_3 & + & 2x_4 & - & 2x_5 & - & x_6 & = & 10 \\
& x_1 & + & x_2 & + & 2x_3 & - & x_4 & + & x_5 & + & x_6 & = & 12 \\
& \multicolumn{11}{l}{x_1, x_2, x_3, x_4, x_5, x_6} & \geq & 0.
\end{array}
$$

Notice that the right-hand-side of the first constraint in $f(A^1, b^1, c^2)$ has been changed to $(1-\lambda)8 + \lambda 6 = 8 - 2\lambda$. For $\lambda = 0$, the optimal solution is

$$x^1 = (0, 0, 0, 36, 14, 34)^{\mathrm{T}}$$

with the optimal objective value of 20, and the corresponding optimal basis is $B = \{4, 5, 6\}$. This basis remains optimal for each $\lambda \in [0, 1]$. Since $\theta(1) = 26 < 50 = \theta$, we proceed to the second stage, where the left-hand side of the first constraint is replaced by a convex combination of A^1_{1*} and A^2_{1*}. We have the following problem:

$$\begin{array}{lrcrcrcrcrcrcl}
\text{min} & -x_1 & + & x_2 & + & 2x_3 & - & x_4 & + & 4x_5 & & & & \\
\text{subject to} & (2+\lambda)x_1 & + & (1+3\lambda)x_2 & + & (2+5\lambda)x_3 & + & x_4 & - & 2x_5 & + & \lambda x_6 & = & 6 \\
 & -x_1 & + & x_2 & + & x_3 & + & 2x_4 & - & 2x_5 & - & x_6 & = & 10 \\
 & x_1 & + & x_2 & + & 2x_3 & - & x_4 & + & x_5 & + & x_6 & = & 12 \\
 & & & & & & & & & \multicolumn{3}{r}{x_1, x_2, x_3, x_4, x_5, x_6} & \geq & 0.
\end{array}$$

For $\lambda \geq 0$, we have

$$A(\lambda)_B = \begin{pmatrix} 1 & -2 & \lambda \\ 2 & -2 & -1 \\ -1 & 1 & 1 \end{pmatrix}.$$

By using the Sherman-Morrison formula for the inverse matrix, we obtain

$$A(\lambda)_B^{-1} = \begin{pmatrix} -1 & \lambda+2 & 2\lambda+2 \\ -1 & \lambda+1 & 2\lambda+1 \\ 0 & 1 & 2 \end{pmatrix}.$$

To obtain the stability region on which this basis is dual feasible (the optimality criterion is satisfied), first we find

$$\begin{aligned}
& c^2_R - c^2_B A(\lambda)_B^{-1} A(\lambda)_R \\
&= (-1, 1, 2) - (-1, 4, 0) \begin{pmatrix} -1 & \lambda+2 & 2\lambda+2 \\ -1 & \lambda+1 & 2\lambda+1 \\ 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} 2+\lambda & 1+3\lambda & 2+5\lambda \\ -1 & 1 & 1 \\ 1 & 1 & 2 \end{pmatrix} \\
&= (5, 0, 2) \\
&\geq 0.
\end{aligned}$$

This implies that dual feasibility holds for each $\lambda \in [0, 1]$. To check the feasibility criterion, we have to calculate

$$A(\lambda)_B^{-1} b = \begin{pmatrix} -1 & \lambda+2 & 2\lambda+2 \\ -1 & \lambda+1 & 2\lambda+1 \\ 0 & 1 & 2 \end{pmatrix} \begin{pmatrix} 6 \\ 10 \\ 12 \end{pmatrix} = \begin{pmatrix} 38+34\lambda \\ 16+34\lambda \\ 34 \end{pmatrix}.$$

For each $\lambda \in [0, 1]$ the current basis is primal feasible. Hence, for each $\lambda \in [0, 1]$ the basis B remains optimal. The optimal objective value $\theta(\lambda)$ on this interval is given by

$$\theta(\lambda) = c_B^2 A(\lambda)_B^{-1} b = (-1, 4, 0) \begin{pmatrix} 38 + 34\lambda \\ 16 + 34\lambda \\ 34 \end{pmatrix} = 26 + 102\lambda.$$

The desired $\lambda \in [0, 1]$ for which $\theta(\lambda) = 50$ is simply $\lambda = \frac{4}{17}$.

Therefore, the problem in the family with the optimal value equal to 50 is

$$\begin{array}{rllcrcrcrcrcrcl} 50 & = & \min & -x_1 & + & x_2 & + & 2x_3 & - & x_4 & + & 4x_5 & & & \\ & & \text{subject to} & \frac{38}{17}x_1 & + & \frac{29}{17}x_2 & + & \frac{54}{17}x_3 & + & x_4 & - & 2x_5 & + & \frac{4}{17}x_6 & = & 6 \\ & & & -x_1 & + & x_2 & + & x_3 & + & 2x_4 & - & 2x_5 & - & x_6 & = & 10 \\ & & & x_1 & + & x_2 & + & 2x_3 & - & x_4 & + & x_5 & + & x_6 & = & 12 \\ & & & \multicolumn{11}{l}{x_1, x_2, x_3, x_4, x_5, x_6} & \geq & 0. \end{array}$$

5. An application and comparison with Binary Search

We continue with the notation of Section 4.4. In particular, (A^1, b^1, c^1) is the scenario under which the lower bound f^L is attained, and (A^2, b^2, c^2) is the scenario under which the upper bound f^U is attained.

The method of Section 4.4 can be understood as a method for step-by-step movement from the point (A^1, b^1, c^1) to the point (A^2, b^2, c^2) in the space $\boldsymbol{A} \times \boldsymbol{b} \times \boldsymbol{c}$. By continuity of f on $\boldsymbol{A} \times \boldsymbol{b} \times \boldsymbol{c}$, we are sure that we pass through every value $\theta \in [f^L, f^U]$. To recall how we make the steps: in the space $\boldsymbol{A} \times \boldsymbol{b} \times \boldsymbol{c}$ we first move in the $\boldsymbol{c}$-subspace in the direction $c^2 - c^1$. Then, for each $k = 1, \ldots, m$, we make steps as follows: for a given k, a walk in the $\boldsymbol{b}$-subspace in the direction b_k ("First Stage") is followed by a walk in the $\boldsymbol{A}$-subspace in the direction $A^2_{k*} - A^1_{k*}$ ("Second Stage").

5.1. A drawback of Binary Search

In Section 4.4 we have already mentioned that it is tempting to go straight from (A^1, b^1, c^1) to (A^2, b^2, c^2) in $\boldsymbol{A} \times \boldsymbol{b} \times \boldsymbol{c}$. Indeed, this can be done with Binary Search, using Bolzano's Intermediate Value Theorem. Given $\theta \in (f^L, f^U)$, define

$$\Theta(\lambda) = f((1-\lambda)A^1 + \lambda A^2, (1-\lambda)b^1 + \lambda b^2, (1-\lambda)c^1 + \lambda c^2) - \theta$$

and run Binary Search over $\lambda \in [0, 1]$. Since $\Theta(0) < 0 < \Theta(1)$, the procedure finds a zero point λ_0 of Θ up to any given precision $\varepsilon > 0$. (Here it is interesting to note that f is a polynomial-time function. Therefore we can expect that one iteration of the Binary Search procedure will be computationally fast.)

The Binary Search procedure has some advantages and some drawbacks. The main problem is that it may happen that the procedure finds a scenario

$$A_0 := (1-\lambda_0)A^1 + \lambda_0 A^2, \quad b_0 := (1-\lambda_0)b^1 + \lambda_0 b^2, \quad c_0 := (1-\lambda_0)c^1 + \lambda_0 c^2$$

such that the value $|f(A_0, b_0, c_0) - \theta|$ is small, but the distance

$$\min_{\lambda \in [0,1]} \{|\lambda_0 - \lambda| : f(A_0^*, b_0^*, c_0^*) = \theta,\ A_0^* = (1-\lambda)A^1 + \lambda A^2,\\ b_0^* = (1-\lambda)b^1 + \lambda b^2,\ c_0^* = (1-\lambda)c^1 + \lambda c^2\}$$

is large. The problem is illustrated by Figure 1. The Binary Step procedure finds an ε-approximate solution (A_0, b_0, c_0) in one step, but the correct scenario (A_0^*, b_0^*, c_0^*) is "far" from (A_0, b_0, c_0).

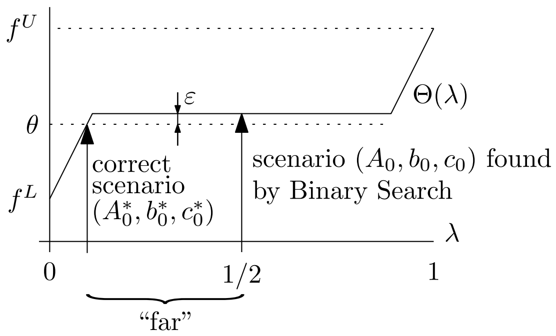

Figure 1: Illustration of possible imprecision of the Binary Search procedure.

Remark 3. The function Θ of Figure 1 is nondecreasing. Observe that this does not hold in general. For example, consider the problem $\max\{x_2 : x_2 \le (1-2\lambda)x_1;\ x_1 \le -1 + 20\lambda;\ x_1 \le 5 - 4\lambda;\ x_1 \ge -10;\ x_2 \ge -10\}$, whose function Θ has a graph with an increasing-decreasing-increasing shape. This shows that it will be difficult to bound the number of steps of the Binary Search procedure, necessary to achieving ε-convergence, a priori.

Remark 4. To achieve the increasing-constant-increasing shape of the function Θ, illustrated by Figure 1, consider the problem $\min\{x_1 + x_2 : x_1 \ge 1 - 2\lambda;\ (10 - 9\lambda)x_1 + 2x_2 \ge 0;\ x_2 \ge 0\}$.

The main advantage of the method of Section 4.4 is that it finds a scenario (A_0^*, b_0^*, c_0^*), such that $f(A_0^*, b_0^*, c_0^*) = \theta$, exactly. Nevertheless, both methods — the method of Section 4.4 and Binary Search — are complementary. In the next section we will illustrate when either the first or the latter can be chosen and how their results differ.

Complexity. As for computational complexity, neither of the methods is a winner, showing their complementarity again. Binary Search usually converges within fewer iterations, but the work in one iteration is greater since a full linear program must be solved. On the other hand, the parametric programming technique usually requires more iterations, but one iteration just amounts to a basis switch.

5.2. Example: A matrix casino

Consider a zero-sum matrix game with the payoff matrix A. The columns represent the strategies of Player I and the rows represent the strategies of Player II. It is well known that for finding the Nash mixed strategy for Player I it suffices to solve the linear program

$$\max \gamma \text{ subject to } A^{\mathrm{T}}x \geq \gamma e,\ e^{\mathrm{T}}x = 1,\ x \geq 0.$$

The optimal value γ^* is the value of the game — it is the best achievable average win/loss when both players obey their Nash strategies.

A matrix casino is a casino where matrix games of chance are played. We will consider a well-known and ancient game, called Morra [39]. To recall: each player throws out a hand, showing from one to five fingers, and calls out loud his guess on the number of fingers shown by the other player. If both players guess right or none of them does, nobody wins and nothing is paid. If only one player guesses right, he wins an amount of dollars equal to the sum of fingers shown out by both players. (Here we write "he" instead of the usual gender-correct expression "(s)he" since a woman is less likely to be a gambler.)

Casino ↓		Gambler: showed out fingers (S) / guessed fingers (G)																								
		S1					S2					S3					S4					S5				
		G1	G2	G3	G4	G5	G1	G2	G3	G4	G5	G1	G2	G3	G4	G5	G1	G2	G3	G4	G5	G1	G2	G3	G4	G5
S1	G1		−2	−2	−2	−2	3					4					5					6				
	G2	2						−3	−3	−3	−3	4					5					6				
	G3	2					3						−4	−4	−4	−4	5					6				
	G4	2					3					4						−5	−5	−5	−5	6				
	G5	2					3					4					5						−6	−6	−6	−6
S2	G1	−3		−3	−3	−3		4					5					6					7			
	G2		3				−4		−4	−4	−4		5					6					7			
	G3		3					4				−5		−5	−5	−5		6					7			
	G4		3					4					5				−6		−6	−6	−6		7			
	G5		3					4					5					6				−7		−7	−7	−7
S3	G1	−4	−4		−4	−4			5					6					7					8		
	G2			4			−5	−5		−5	−5			6					7					8		
	G3			4					5			−6	−6		−6	−6			7					8		
	G4			4					5					6			−7	−7		−7	−7			8		
	G5			4					5					6					7			−8	−8		−8	−8
S4	G1	−5	−5	−5		−5				6					7					8					9	
	G2				5		−6	−6	−6		−6				7					8					9	
	G3				5					6		−7	−7	−7		−7				8					9	
	G4				5					6					7		−8	−8	−8		−8				9	
	G5				5					6					7					8		−9	−9	−9		−9
S5	G1	−6	−6	−6	−6						7					8					9					10
	G2					6	−7	−7	−7	−7						8					9					10
	G3					6					7	−8	−8	−8	−8						9					10
	G4					6					7					8	−9	−9	−9	−9						10
	G5					6					7					8					9	−10	−10	−10	−10	

Table 1: Payoff matrix of the Morra game to be introduced in the matrix casino.

The game is symmetric for both players and hence its value is zero. Each player has 25 possible strategies: a strategy (S, G) with $S \in \{1, \dots, 5\}$ and $G \in \{1, \dots, 5\}$ means that the player showed out S fingers and guessed that the enemy will show G fingers. The payoff matrix is shown in Table 1.

Now we turn to the formulation of our problem. Our casino wants to attract new gamblers and it decides to introduce Morra as a new game. Say that a gambler acts as Player I and the casino acts as Player II. However, for a casino it is not acceptable to play a zero-valued game. We need to modify the payoff matrix A into a form such that the value of the game becomes slightly negative for the gambler. For example, we might need to adjust A to achieve the value of the game $\theta := -\frac{1}{37}$. This is reasonable when we want to have, on average, the same profit as from roulette.[1] If we introduced a new game in the casino, with a different value than roulette has, players would systematically prefer the game with the better value and this would be undesirable.

We will proceed as follows. We allow a perturbation of the elements of the payoff matrix A and find a scenario attaining θ. It is natural to consider (at least) the following possibilities:

(a) *Absolute perturbations of nonzero elements:* each nonzero element a_{ij} of A is allowed to be perturbed by a fixed amount $\delta > 0$. That is, we set $a^L_{ij} = a_{ij} - \delta$ and $a^U_{ij} = a_{ij} + \delta$ if $a_{ij} \neq 0$ and $a^L_{ij} = a^U_{ij} = 0$ otherwise.

(b) *Relative perturbations:* each element a_{ij} of A is allowed to be perturbed by a given relative amount $0 < \varrho \leq 1$. That is, we set $a^L_{ij} = (1 - \varrho)a_{ij}$, $a^U_{ij} = (1 + \varrho)a_{ij}$ for $a_{ij} \geq 0$, and $a^L_{ij} = (1 + \varrho)a_{ij}$, $a^U_{ij} = (1 - \varrho)a_{ij}$ for $a_{ij} < 0$.

In the sequel, we will continue with the approach (a) only; the analysis of the case (b) would be analogous.

So we are to find a matrix $A_0 \in [A^L, A^U]$ such that

$$\theta = \max\{\gamma : A_0^{\mathrm{T}} x \geq \gamma e,\ e^{\mathrm{T}} x = 1,\ x \geq 0\}. \tag{28}$$

Remark 5. Formally, the method of Section 4.4 requires a linear program in the form $\min\{\widetilde{c}x : \widetilde{A}x = \widetilde{b}, x \geq 0\}$. We can use a traditional trick: we choose a sufficiently large number $\kappa > 0$, say $\kappa = 1 + \max_{i,j} |A_{ij}|$, and

$$\text{replace } A^L, A^U \text{ by } A^L + \kappa e e^{\mathrm{T}}, A^U + \kappa e e^{\mathrm{T}}, \text{ respectively.} \tag{29}$$

This assures that for every admissible A, the value of the game is nonnegative, and hence we can add the constraint $\gamma \geq 0$. We can also add slacks for the inequalities and obtain the equality form with all variables nonnegative. We used the fact that addition of a constant κ to each element of A increases the value of the game by κ.

[1] To be more precise, say that we have the roulette with numbers $0, \dots, 36$, where 0 is green and $1, \dots, 36$ are red and black. For simplicity say that a player can only bet red or black. Due to the presence of the green zero, the value is indeed $-\frac{1}{37}$.

The trick (29) also allows us to prove the following proposition which guarantees continuity of the objective value function $f(A, b, c)$.

Proposition 2. *The conditions (7) and (8) are satisfied.*

Proof. By the trick (29) we may assume that $\gamma \geq 0$. The linear program (28) can be written as

$$\min \left\{ (-1, 0, 0) \begin{pmatrix} \gamma \\ x \\ s \end{pmatrix} : \begin{pmatrix} e & -A^{\mathrm{T}} & I \\ 0 & e^{\mathrm{T}} & 0 \end{pmatrix} \begin{pmatrix} \gamma \\ x \\ s \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad \begin{pmatrix} \gamma \\ x \\ s \end{pmatrix} \geq 0 \right\}.$$

Then, system (9) has the form

$$\gamma e - A^{\mathrm{T}} x + s = 0,\ e^{\mathrm{T}} x = 0,\ \gamma \geq 0,\ x \geq 0,\ s \geq 0,\ -\gamma \leq 0,\ \gamma + e^{\mathrm{T}} x + e^{\mathrm{T}} s = 1.$$

Since $x \geq 0$ and $e^{\mathrm{T}} x = 0$, we have $x = 0$, and the first equality states that $s = -\gamma e$. Since $\gamma \geq 0$ and $s \geq 0$, we have $s = 0$ and $\gamma = 0$. Then $\gamma + e^{\mathrm{T}} x + e^{\mathrm{T}} s = 0 \neq 1$. Hence (7) holds true.

To prove (8), observe that system (11) has the form

$$e^{\mathrm{T}} y_1 \leq 0,\ -Ay_1 + y_2 e \leq 0,\ y_1 \leq 0,\ y_2 \geq 0,\ \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} \neq 0.$$

By the trick (29) we may assume that $A^L \geq 0$. Then also $A \geq 0$. Since $y_1 \leq 0$, we have $-Ay_1 \geq 0$. Thus we have $0 \leq -Ay_1 \leq -y_2 e \leq 0e = 0$, implying that $y_1 = 0$ and $y_2 = 0$. □

In this example we will compare the results obtained by the method of Section 4.4 and the Binary Search procedure. Since there are intervals neither in the objective function nor in the right-hand sides, only the "Second Stage" of the method of Section 4.4 is applicable.

Observe that a positive perturbation of any element of A cannot decrease the value of the game. It follows that f^L is attained in A^L and f^U is attained in A^U, so both extremal optimal values and the corresponding scenarios are computable efficiently.

The method of Section 4.4 perturbs the lines of A^{T} one by one, until the demand $\theta = -\frac{1}{37}$ is attained. To visualize the trajectory of the method, we define the function $\theta^*(\mu)$ with $\mu \in [1, 26]$ as follows. Given $k \in \{1, \ldots, 25\}$ and $\lambda \in [0, 1]$, let $A_k(\lambda)$ denote the matrix defined by (24), (25), (26) with $A^1 = (A^L)^{\mathrm{T}}$ and $A^2 = (A^U)^{\mathrm{T}}$. The function

$$\theta_k(\lambda) = \max\{\gamma : A_k(\lambda)^{\mathrm{T}} x \geq \gamma e,\ e^{\mathrm{T}} x = 1,\ x \geq 0\}, \quad \lambda \in [0, 1]$$

describes the changes in the value of our game when k-th line of A^{T} is being λ-perturbed. By continuity, we have $\theta_k(1) = \theta_{k+1}(0)$. It is natural to join the graphs of $\theta_1, \theta_2, \ldots, \theta_{25}$ and define the function

$$\theta^*(\mu) := \theta_{\lfloor \mu \rfloor}(\mu - \lfloor \mu \rfloor), \quad \mu \in [1, 26].$$

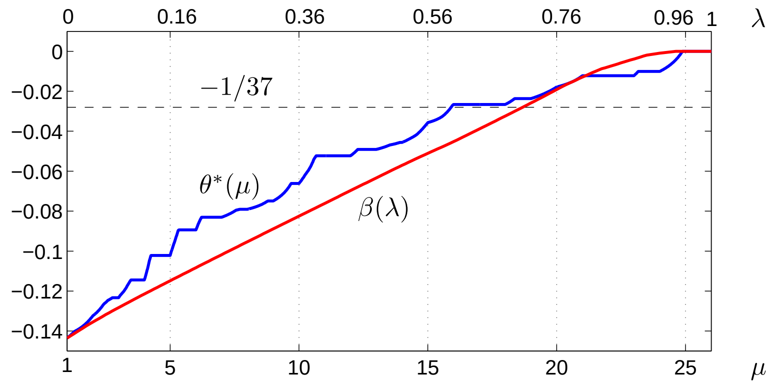

Figure 2: The function $\theta^*(\mu)$ of the method of Section 4.4 and the function $\beta(\lambda)$ of Binary Search.

We have $\theta^*(1) = f^L$ and $\theta^*(26) = f^U$. For example, the fact $\theta^*(8.4) = \theta$ tells us that the desired value of the game has been achieved when the procedure was processing 8th line of A^{T} with perturbation $\lambda = 0.4$.

In this example we consider absolute perturbations of nonzero elements with the tolerance rate $\delta = 0.15$. Figure 2 shows the function θ^* and its comparison with the "trajectory" of Binary Search defined as

$$\beta(\lambda) = \max\{\gamma : (1-\lambda)(A^L)^{\mathrm{T}}x + \lambda(A^U)^{\mathrm{T}}x \geq \gamma e,\ e^{\mathrm{T}}x = 1,\ x \geq 0\}, \quad \lambda \in [0,1].$$

The resulting perturbed payoff matrices are shown in Tables 2 and 3.

Casino ↓		Gambler: showed out fingers (S) / guessed fingers (G)																								
		S1					S2					S3					S4					S5				
		G1	G2	G3	G4	G5	G1	G2	G3	G4	G5	G1	G2	G3	G4	G5	G1	G2	G3	G4	G5	G1	G2	G3	G4	G5
	G1		−2	−2	−2	−2	3					4					4.5					5.5				
	G2	2						−3	−3	−3	−3	4					4.5					5.5				
S1	G3	2					3						−4	−4	−4	−4.015	4.5					5.5				
	G4	2					3					4						−5.5	−5.5	−5.5	−5.5	5.5				
	G5	2					3					4					4.5						−6.5	−6.5	−6.5	−6.5
	G1	−3		−3	−3	−3		4					5					5.5					6.5			
	G2		3				−4		−4	−4	−4		5					5.5					6.5			
S2	G3		3					4				−5		−5	−5	−5.015		5.5					6.5			
	G4		3					4					5				−6.5		−6.5	−6.5	−6.5		6.5			
	G5		3					4					5					5.5				−7.5		−7.5	−7.5	−7.5
	G1	−4	−4		−4	−4			5					6					6.5					7.5		
	G2			4			−5	−5		−5	−5			6					6.5					7.5		
S3	G3			4					5			−6	−6		−6	−6.015			6.5					7.5		
	G4			4					5					6			−7.5	−7.5		−7.5	−7.5			7.5		
	G5			4					5					6					6.5			−8.5	−8.5		−8.5	−8
	G1	−5	−5	−5		−5				6					7					7.5					8.5	
	G2				5		−6	−6	−6		−6				7					7.5					8.5	
S4	G3				5					6		−7	−7	−7		−7.015				7.5					8.5	
	G4				5					6					7		−8.5	−8.5	−8.5		−8.5				8.5	
	G5				5					6					7					7.5		−9.5	−9.5	−9.5		−9.5
	G1	−6	−6	−6	−6						7					7.985					8.5					9.5
	G2					6	−7	−7	−7	−7						7.985					8.5					9.5
S5	G3					6					7	−8	−8	−8	−8	7.985					8.5					9.5
	G4					6					7					7.985	−9.5	−9.5	−9.5	−9.5						9.5
	G5					6					7					7.985					8.5	−10.5	−10.5	−10.5	−10.5	

Table 2: Perturbed payoff matrix A: the result of the method of Section 4.4. The procedure terminated when perturbing the strategy S3G5.

Casino		Gambler: showed out fingers (S) / guessed fingers (G)																								
↓		S1					S2					S3					S4					S5				
		G1	G2	G3	G4	G5	G1	G2	G3	G4	G5	G1	G2	G3	G4	G5	G1	G2	G3	G4	G5	G1	G2	G3	G4	G5
	G1		−2.14	−2.14	−2.14	−2.14	2.86					3.86					4.86					5.86				
	G2	1.86						−3.14	−3.14	−3.14	−3.14	3.86					4.86					5.86				
S1	G3	1.86					2.86						−4.14	−4.14	−4.14	−4.14	4.86					5.86				
	G4	1.86					2.86					3.86						−5.14	−5.14	−5.14	−5.14	5.86				
	G5	1.86					2.86					3.86					4.86						−6.14	−6.14	−6.14	−6.14
	G1	−3.14		−3.14	−3.14	−3.14		3.86					4.86					5.86					6.86			
	G2		2.86				−4.14		−4.14	−4.14	−4.14		4.86					5.86					6.86			
S2	G3		2.86					3.86				−5.14		−5.14	−5.14	−5.14		5.86					6.86			
	G4		2.86					3.86					4.86				−6.14		−6.14	−6.14	−6.14		6.86			
	G5		2.86					3.86					4.86					5.86				−7.14		−7.14	−7.14	−7.14
	G1	−4.14	−4.14		−4.14	−4.14			4.86					5.86					6.86					7.86		
	G2			3.86			−5.14	−5.14		−5.14	−5.14			5.86					6.86					7.86		
S3	G3			3.86					4.86			−6.14	−6.14		−6.14	−6.14			6.86					7.86		
	G4			3.86					4.86					5.86			−7.14	−7.14		−7.14	−7.14			7.86		
	G5			3.86					4.86					5.86					6.86			−8.14	−8.14		−8.14	−8.14
	G1	−5.14	−5.14	−5.14		−5.14				5.86					6.86					7.86					8.86	
	G2				4.86		−6.14	−6.14	−6.14		−6.14				6.86					7.86					8.86	
S4	G3				4.86					5.86		−7.14	−7.14	−7.14		−7.14				7.86					8.86	
	G4				4.86					5.86					6.86		−8.14	−8.14	−8.14		−8.14				8.86	
	G5				4.86					5.86					6.86					7.86		−9.14	−9.14	−9.14		−9.14
	G1	−6.14	−6.14	−6.14	−6.14						6.86					7.86					8.86					9.86
	G2					5.86	−7.14	−7.14	−7.14	−7.14						7.86					8.86					9.86
S5	G3					5.86					6.86	−8.14	−8.14	−8.14	−8.14						8.86					9.86
	G4					5.86					6.86					7.86	−9.14	−9.14	−9.14	−9.14						9.86
	G5					5.86					6.86					7.86					8.86	−10.14	−10.14	−10.14	−10.14	

Table 3: The perturbed payoff matrix A using Binary Search.

We summarize the main differences between the two methods. Binary Search processes the matrix A^{T} globally — it perturbs all elements simultaneously. On the other hand, the method of Section 4.4 makes the perturbation strategy-by-strategy: it attempts to perturb payoffs of as few strategies as possible. Hence we can say that it attempts to process the matrix A^{T} locally. Moreover, the method of Section 4.4 does not rely on a particular ordering of rows of A^{T}. Hence the rows of A^{T} can be permuted. This is useful when there are strategies the payoffs of which are more preferred for perturbation than other strategies, the payoffs of which are preferred to be kept unchanged.

5.3. Example: How to determine a fee for playing a game in the matrix casino

Here we continue with the example of the previous section.

There is another interesting strategy to achieve the value $-\frac{1}{37}$ of Morra. The gambler is to pay a fee for a game in which nobody wins. (Observe that introduction of the fee can change the pair of Nash strategies.) So we are to perturb the *zero* elements of A slightly by the same amount. Here we can take the advantage of the Binary Search technique that it perturbs all elements simultaneously. In particular, we define $A^U = A$ and

$$a_{ij}^L = \begin{cases} -\Delta & \text{if } a_{ij} = 0, \\ a_{ij} & \text{if } a_{ij} \neq 0, \end{cases}$$

where Δ is a sufficiently large number chosen in advance, such that $f^L < -\frac{1}{37}$. Now the Binary Search will perturb all zero elements of A identically. We find out that the value of the fee is 0.0263. Note that this value is different from $\frac{1}{37} = 0.0270$.

6. Conclusions

We designed a new method for solving the inverse interval LP problem when the coefficients can be selected from given intervals. This is interesting when the coefficients of the LP act as "controlling variables" and the optimal value is prescribed. There are various examples — we illustrated the approach by the Matrix Casino example, where the task is to design a suitable matrix game. More precisely, the task is to find a payoff matrix from a given neighborhood of a given matrix such that the resulting game has a prescribed value. There are many more examples; for instance, we can seek for a network with a prescribed maximal flow, when the capacities of edges can be chosen from given intervals.

To find the appropriate scenario, we employed and extended parametric analysis concepts of LP. Our approach also provides a new connection between inverse optimization and parametric analysis in LP theory.

We also compared our method with Binary Search. The main drawback of Binary Search is that is finds only an ε-approximate solution, and the optimal one can be far away. On the other hand, from a practical perspective, Binary Search often gives a good estimate in only a small number of iterations (unless the optimal value function is "wild"). Thus, an efficient combination of both approaches may be promising and deserves to be a subject of the future research.

It is also interesting to reformulate the method for linear programming problems when the coefficients need not be intervals, but polytopes, or — more generally — compact convex sets.

The solution of the inverse LP problem need not be unique. Hence it is also interesting to ask which of the solutions is "better" or "worse" and give a precise definition of what "being a better solution" exactly means. (Such a definition may differ problem by problem.) Of course, it is then natural to adapt the method for finding the "better" solutions.

References

[1] B. Machost, Numerische Behandlung des Simplexverfahrens mit intervallanalytischen Methoden, Tech. Rep. 30, Berichte der Gesellschaft für Mathematik und Datenverarbeitung, 54 pages, Bonn, 1970.

[2] R. Krawczyk, Fehlerabschätzung bei linearer Optimierung, in: Interval Mathematics, LNCS 29, Springer, Berlin, 215–222, 1975.

[3] H. Beeck, Linear programming with inexact data, technical report TUM-ISU-7830, Technical University of Munich, Munich, 1978.

[4] J. W. Chinneck, K. Ramadan, Linear programming with interval coefficients, J. Oper. Res. Soc. 51 (2) (2000) 209–220.

[5] M. Fiedler, J. Nedoma, J. Ramík, J. Rohn, K. Zimmermann, Linear optimization problems with inexact data, Springer, New York, 2006.

[6] M. Hladík, Optimal value range in interval linear programming, Fuzzy Optim. Decis. Mak. 8 (3) (2009) 283–294.

[7] M. Hladík, Interval linear programming: A survey, in: Z. A. Mann (Ed.), Linear Programming - New Frontiers in Theory and Applications, chap. 2, Nova Science Publishers, New York, 85–120, 2012.

[8] D.-F. Li, Linear programming approach to solve interval-valued matrix games, Omega 39 (6) (2011) 655–666.

[9] F. Mráz, The exact lower bound of optimal values in interval LP, in: G. Alefeld, A. Frommer, B. Lang (Eds.), Scientific Computing and Validated Numerics, Akademie Velag, Berlin, 1996.

[10] F. Mráz, Calculating the exact bounds of optimal values in LP with interval coefficients, Ann. Oper. Res. 81 (1998) 51–62.

[11] M. Allahdadi, H. M. Nehi, The optimal solution set of the interval linear programming problems, Optim. Lett. In press, DOI 10.1007/s11590-012-0530-4.

[12] M. Hladík, How to determine basis stability in interval linear programming, Optim. Lett. In press, DOI: 10.1007/s11590-012-0589-y.

[13] C. Jansson, A self-validating method for solving linear programming problems with interval input data, in: U. Kulisch, H. J. Stetter (Eds.), Scientific computation with automatic result verification, Computing Suppl. 6, Springer, Wien, 33–45, 1988.

[14] C. Jansson, On self-validating methods for optimization problems, in: J. Herzberger (Ed.), Topics in validated computations, Elsevier, Amsterdam, 381–438, 1994.

[15] C. Jansson, Calculation of exact bounds for the solution set of linear interval systems, Linear Algebra Appl. 251 (1997) 321–340.

[16] C. Jansson, S. M. Rump, Rigorous solution of linear programming problems with uncertain d ata, Z. Oper. Res. 35 (2) (1991) 87–111.

[17] M. Hladík, Solution set characterization of linear interval systems with a specific dependence structure, Reliab. Comput. 13 (4) (2007) 361–374.

[18] M. S. Bazaraa, J. J. Jarvis, H. D. Sherali, Linear programming and network flows. 4th ed., John Wiley & Sons, New York, 2010.

[19] T. Gal, Postoptimal analyses, parametric programming, and related topics, McGraw-Hill, New York, 1979.

[20] T. Gal, H. J. Greenberg (Eds.), Advances in sensitivity analysis and parametric programming, Kluwer Academic Publishers, Boston, 1997.

[21] Nožička, František and Guddat, Jürgen and Hollatz, Horst and Bank, Bernd, Theorie der linearen parametrischen Optimierung, Akademie-Verlag, Berlin, 1974.

[22] P. Yu, M. Zeleny, Linear multiparametric programming by multicriteria simplex method, Manage. Sci. 23 (1976) 159–170.

[23] S. Ahmed, Y. Guan, The inverse optimal value problem, Math. Program., Ser. A 102 (2005) 91–110.

[24] R. K. Ahuja, J. B. Orlin, Inverse optimization, Oper. Res. 49 (5) (2001) 771–783.

[25] X.-X. Z. L. Jiang, Yong, J. Zhang, A Perturbation approach for a type of inverse linear programming problems, Int. J. Comp. Math. 88 (3) (2011) 508–516.

[26] J. Zhang, Z. Liu, Calculating some inverse linear programming problems, J. Comput. Appl. Math. 72 (2) (1996) 261–273.

[27] J. Zhang, Z. Liu, A further study on inverse linear programming problems., J. Comput. Appl. Math. 106 (2) (1999) 345–359.

[28] A. J. Schaefer, Inverse integer programming, Optim. Lett. 3 (4) (2009) 483–489.

[29] Z. Duan, L. Wang, Heuristic algorithms for the inverse mixed integer linear programming problem, J. Glob. Optim. 51 (3) (2011) 463–471.

[30] M. Hladík, Tolerances in portfolio selection via interval linear programming, in: CD-ROM Proceedings 26-th Int. Conf. Mathematical Methods in Economics MME08, Liberec, Czech Republic, 185–191, 2008.

[31] M. Vranka, Interval Linear Programming, Master's thesis, Faculty of Mathematics and Physics, Charles University in Prague, 2005.

[32] R. J.-B. Wets, On the continuity of the value of a linear program and of related polyhedral-valued multifunctions, Math. Program. Study 24 (1985) 14–29.

[33] M. Hladík, Weak and strong solvability of interval linear systems of equations and inequalities, Linear Algebra Appl. 438 (11) (2013) 4156–4165.

[34] A. G. Hadigheh, K. Mirnia, T. Terlaky, Active constraint set invariancy sensitivity analysis in linear optimization, J. Optim. Theory Appl. 133 (3) (2007) 303–315.

[35] A. G. Hadigheh, T. Terlaky, Generalized support set invariancy sensitivity analysis in linear optimization, J. Ind. Manag. Optim. 2 (1) (2006) 1–18.

[36] A. G. Hadigheh, T. Terlaky, Sensitivity analysis in linear optimization: invariant support set intervals, Eur. J. Oper. Res. 169 (3) (2006) 1158–1175.

[37] M. Hladík, Multiparametric linear programming: support set and optimal partition invariancy, Eur. J. Oper. Res. 202 (1) (2010) 25–31.

[38] L. Grygarová, Die Lösbarkeit eines linearen Optimierungsproblems unter Zufügung einer weiteren Restriktionsbedingung., Apl. Mat. 17 (1972) 352–387.

[39] J. Anděl, Mathematics of chance, Wiley, New York, first edn., 2001.