UNIVERSITYOF BIRMINGHAM

Tropical pseudolinear and pseudoquadratic optimization as parametric mean-payoff games

Parsons, Jamie; Sergeev, Sergei; Wang, Huili

Document Version
Publisher's PDF, also known as Version of record

[Link to publication on Research at Birmingham portal](Link)

Tropical pseudolinear and pseudoquadratic optimization as parametric mean-payoff games

Jamie Parsons, Sergei Sergeev & Huili Wang

Published online: 29 Jun 2022.

Submit your article to this journal ↗

Article views: 92

View related articles ↗

View Crossmark data ↗

Tropical pseudolinear and pseudoquadratic optimization as parametric mean-payoff games

Jamie Parsons[a], Sergei Sergeev[a] and Huili Wang[b]

[a]School of Mathematics, University of Birmingham, Edgbaston, UK; [b]School of Sciences, Southwest Petroleum University, Chengdu, Sichuan, People's Republic of China

ABSTRACT

We apply an approach based on parametric mean-payoff games to develop bisection and Newton schemes for solving problems of tropical pseudolinear and pseudoquadratic optimization with general two-sided constraints.

1. Introduction

Tropical linear algebra is a relatively new area of mathematics, having only been studied in depth since around 1960. Cuninghame-Green was one of the pioneers in the area, with many of his results being presented in [1]. Since then, a number of mathematicians have further developed the theory and applications of tropical linearity, and it has enjoyed prominence and influence in several mathematical areas such as linear algebra, algebraic geometry and combinatorial optimization. Tropical linear algebra gives us the ability to write seemingly non-linear problems in a linear fashion using tropical linear operators, allowing an algebraic encoding of many combinatorial problems and providing a framework with which some discrete problems can be modelled [2,3].

Optimization problems over tropical linear algebra have been studied since 1970's and 1980's: see, e.g. K. Zimmermann [4] and U. Zimmermann [5]. A crucial step was made by Butkovič [3], who developed a bisection scheme for solving tropical linear programming. Krivulin [6–12] made a big contribution to the area by solving a multitude of what we term as tropical pseudolinear and tropical pseudoquadratic optimization problems, albeit with special constraints. The connection between tropical linear algebra and mean payoff games was discovered by Akian et al. [13], and this connection was further applied to tropical linear-fractional programming by Gaubert et al. [14]. The present paper aims to develop the tropical pseudolinear and pseudoquadratic optimization by

CONTACT Sergei Sergeev   s.sergeev@bham.ac.uk

considering a more general form of constraints and applying to it the connection to mean-payoff games discovered in [13], following the approach of [14].

We now introduce the tropical linear algebra formally. Tropical semiring is the set $\mathbb{R}_{\max} = \mathbb{R} \cup \{-\infty\}$ equipped with the operations $(\oplus, \otimes)$ defined by $a \oplus b = \max(a, b)$ and $a \otimes b = a + b$. Algebraically, $(\mathbb{R}_{\max}, \oplus, \otimes)$ is a commutative idempotent semifield, meaning in particular that $\oplus$ and $\otimes$ are commutative, associative and satisfy the distributive law. Note that the $\oplus$ identity is $-\infty$ and the $\otimes$ identity is 0. Many of the tools and operations used in usual linear algebra can be applied in tropical linear algebra. An important difference with rings is that $(\mathbb{R} \cup \{-\infty\}, \oplus)$ does not form a group and thus there is no straightforward subtraction in tropical algebra. Symmetrization over tropical algebra has been known for a long time [2], being a kind of substitute for this shortcoming. Also we can use the idempotent property of $\oplus$: for all $x \in \mathbb{R}_{\max} x \oplus x = x$.

The definition of $\oplus$ and $\otimes$ is then extended to include matrices and vectors. Suppose that $A = (a_{ij})$, $B = (b_{ij})$, $C = (c_{ij})$, where $a_{ij}, b_{ij}, c_{ij} \in \mathbb{R}_{\max}$. When A and B are of the same size, $C = A \oplus B$ if $c_{ij} = a_{ij} \oplus b_{ij}$ for all i, j. For matrices A and B of compatible size, $D = A \otimes B$ if $d_{ij} = \bigoplus a_{ik} \otimes b_{kj} = \max_k(a_{ik} + b_{kj})$.

The unit matrix, I, is defined as a square matrix whose diagonal entries are 0, with all off-diagonal entries being $-\infty$.

For $a \in \mathbb{R} \cup \{-\infty\} \cup \{+\infty\}$ we define its conjugate a^- as $-a$ if $a \in \mathbb{R}$, $+\infty$ if $a = -\infty$, and $-\infty$ if $a = +\infty$. This definition is easily extended to vectors: for a column vector $x = (x_i)$ its conjugate is defined as a row vector $x^- = (x_i^-)$.

The ideas and methods of tropical linear algebra and mean-payoff games will help us to solve the following main problems considered in this paper.

Problem 1.1 (Pseudolinear optimization with two-sided constraints): Given $p \in \mathbb{R}_{\max}^n$, $q \in (\mathbb{R} \cup \{+\infty\})^n$, $U, V \in \mathbb{R}_{\max}^{m \times n}$, $b, d \in \mathbb{R}_{\max}^m$, find

$$\begin{aligned} &\min_{x \in \mathbb{R}^n} && x^- \otimes p \oplus q^- \otimes x \\ &\text{subject to} && U \otimes x \oplus b \leq V \otimes x \oplus d. \end{aligned}$$

and at least one finite $x \in \mathbb{R}^n$ that attains the minimum.

Problem 1.2 (Pseudoquadratic optimization with two-sided constraints): Given $C \in \mathbb{R}_{\max}^{n \times n}$, $p \in \mathbb{R}_{\max}^n$, $q \in (\mathbb{R} \cup \{+\infty\})^n$, $U, V \in \mathbb{R}_{\max}^{m \times n}$, $b, d \in \mathbb{R}_{\max}^m$, find

$$\begin{aligned} &\min_{x \in \mathbb{R}^n} && x^- \otimes C \otimes x \oplus x^- \otimes p \oplus q^- \otimes x \\ &\text{subject to} && U \otimes x \oplus b \leq V \otimes x \oplus d. \end{aligned}$$

and at least one finite $x \in \mathbb{R}^n$ that attains the minimum.

To our knowledge, these problems have not been considered before, and their algorithmic solution based on parametric mean-payoff games, inspired by the solution given by Gaubert et al. [14] for the tropical linear-fractional programming, will be suggested in the present paper for the first time. The problems

with the same objective function as Problem 1.1 but more special constraints were studied by Krivulin [7,12] see also [6] and [15] where the objective fuction is strongly related to the one in Problem 1.1. Problems with similar objective functions as in Problem 1.2 (in particular, containing the pseudoquadratic term $x^- \otimes C \otimes x$) were also studied by Krivulin [8,11,16]. However, the problems considered in these works have more special constraints than Problems 1.1 and 1.2. In return for less general constraints, a comprehensive and concise description of both optimal value and the whole solution set is offered in all of the above references.

In contrast to that approach, we are not interested to describe all solutions. Although it is possible to achieve such a description following, for example, the double description algorithm by Allamigeon et al. [17], it is much more complicated than in the case of special systems of constraints, and we are not intending to do it here.

As in the case of the tropical linear or tropical linear-fractional programming [3,14], there is a clear geometric motivation to consider the problems posed above, since the constraint $U \otimes x \oplus b \leq V \otimes x \oplus d$ is a general form of two-sided tropical affine constraints, that is, such constraints that describe the tropical polyhedra.

We now show how this type of pseudo-quadratic optimization problem may arise in practice, by considering an application to project scheduling, similar to the one described by Krivulin [8,11,16], but also including disjunctive constraints, which were not considered in those works.

Suppose a particular project involves a set of n activities that have to be completed. Each activity has an initiation time x_i and a completion time y_i. We define the time it takes to complete each activity to be its flow time, given by $y_i - x_i$.

We have 'start-to-finish' constraints, that activities cannot be completed until specified times have elapsed after the initiation of other constraints, and that the activities are completed as soon as possible within these constraints. This means that x and y should satisfy the inequalities $x_j + c_{ij} \leq y_i$ for all i and j, where $c_{ij} = -\infty$ if there is no such constraint for some i and j. For the activity j to start as soon as possible, we should have $\max_j(c_{ij} + x_j) = y_i$, and hence we obtain $y_i - x_i = -x_i + \max_j(c_{ij} + x_j)$ for the flow time of task i. We will be also interested to minimize $p_i - x_i$ and $x_i - q_i$ for each task i, which means that we would like task i not to delay too much after q_i and not to start too much in advance before p_i. Note that in the project scheduling practice, one is interested either in minimizing the greatest flow times or in minimizing the greatest of the above mentioned time differences. However, it will be more mathematically convenient for us to consider them together and pose the objective to minimize the greatest of all flow times and the time differences $p_i - x_i$ and $x_i - q_i$. Thus we obtain the objective

$$\min_{x \in \mathbb{R}^n} x^- \otimes C \otimes x \oplus x^- \otimes p \oplus q^- \otimes x.$$

This objective allows us to consider both kinds of objectives at the same time, as we can allow some (or even all) entries of C, p or q^- to be $-\infty$.

The starting times of the tasks may be subject to more constraints: in the simplest case it may be required that $b_i \leq x_i \leq d_i$ for some i and some $b_i, d_i \in \mathbb{R}$, or that $u_{ij} + x_j \leq x_i$ for some i and j and $u_{ij} \in \mathbb{R}$. However, we may also have *disjunctive constraints*, where $b_i \leq v_{ij} + x_j$, for some i and with $b_i, v_{ij} \in \mathbb{R}$, should hold at least for one j. Similarly, we may have that $\max_k(u_{ik} + x_k) \leq v_{ij} + x_j$, for some i with $u_{ik} \in \mathbb{R}_{\max}$ and $v_{ij} \in \mathbb{R}$, should hold for at least one j. This motivates considering the above objective function with constraint in the form of two-sided tropical affine inequality

$$U \otimes x \oplus b \leq V \otimes x \oplus d,$$

which can capture all the above mentioned constraints.

The rest of the paper will be organized as follows. In Section 2 we recall the basic knowledge and facts about tropical linear algebra and mean-payoff games that are needed for this paper. In Section 3 we formulate Problem 1.1 in terms of the associated parametric mean-payoff game and present the certificates of optimality and unboundedness in terms of this game. In Sections 4 and 5 we develop the bisection and Newton schemes for solving Problem 1.1. Two versions of the Newton scheme are given: one for the case of real valued data and the other for the case of integer data. In Section 6 we give an example and describe the numerical experiments, which we conducted upon implementing the Newton and bisection schemes in MATLAB. Finally, in Section 7 we explain how most of our results, including the Newton and bisection schemes, extend to pseudoquadratic programming (Problem 1.2). However, efficient implementation of these schemes for this problem will be developed in another publication.

2. Preliminaries

2.1. Tropical linear algebra

Some of the main concepts of tropical linear algebra come from combinatorial optimization [3].

With a matrix $A \in (\mathbb{R} \cup \{-\infty\})^{n \times n}$ we can associate a weighted digraph $D_A = (N, E, w)$. It has the set of nodes $N = \{1, \ldots, n\}$, set of arcs $E \subseteq N \times N$ such that $(i, j) \in E$ if and only if $a_{ij} \neq -\infty$, and weight function $w : E \to R$ defined by $w(i, j) = a_{ij}$. Vice versa, if we are given a weighted digraph D with n nodes, we can associate with it a matrix $A_D \in (\mathbb{R} \cup \{-\infty\})^{n \times n}$, along the same lines.

One important player in tropical linear algebra is the maximum cycle mean, $\rho(A)$. Given a matrix $A \in (\mathbb{R} \cup -\infty)^{n \times n}$, we define

$$\rho(A) = \max_{\sigma} \frac{w(A, \sigma)}{l(\sigma)}.$$

Here, σ denotes an elementary cycle. Also if we have $\sigma = (i_1 \ldots i_k i_1)$ then $w(A, \sigma) = a_{i_1 i_2} \otimes a_{i_2 i_3} \otimes \ldots \otimes a_{i_k i_1}$ is the weight of $\sigma = (i_1 \ldots i_k i_1)$ and $l(\sigma) = k$ is the length of the cycle σ. Note that $\rho(A)$ is well defined for any matrix A, and $\rho(A) = -\infty$ if and only if D_A is acyclic.

The importance of $\rho(A)$ is due to the following. On the one hand, in the usual linear algebra, the behaviour of some iterative algorithms crucially depends on the existence and properties of $(I - A)^{-1}$. In the tropical linear algebra, this is replaced with the Kleene star defined by the formal series

$$A^* = I \oplus A \oplus A^2 \oplus \ldots.$$

This series converges and is equal to $I \oplus A \oplus \ldots \oplus A^{n-1}$ if and only if $\rho(A) \leq 0$, that is, if and only if there is no cycle in D_A with a positive weight.

On the other hand, $\rho(A)$ is crucial for the tropical eigenvector-eigenvalue problem, or spectral problem in tropical linear algebra, which is the problem of finding tropical eigenvalue λ and tropical eigenvector $x \neq -\infty$ such that $A \otimes x = \lambda \otimes x$. The connection between $\rho(A)$ and the tropical spectral problem is that the greatest tropical eigenvalue of any square matrix over $\mathbb{R}_{\max}$ is equal to $\rho(A)$, see [2,3].

2.2. Dual operators and conjugates

In tropical matrix algebra, matrix inverses exist only for a very special case of diagonal and monomial matrices. However, we can overcome some of the difficulties this poses by defining conjugates.

Dual operations are defined using the min plus algebra, which is the set $\mathbb{R}_{\min} = \mathbb{R} \cup \{+\infty\}$ equipped with operations $(\oplus', \otimes')$ defined by $a \oplus' b = \min(a, b)$ and $a \otimes' b = a + b$ for all $a, b \in \mathbb{R}_{\min}$.

Recalling the definition of scalar conjugate $a \to a^-$, the conjugate of a matrix A with entries in $\mathbb{R} \cup \{-\infty\}$, denoted $A^\sharp$, can be then defined by

$$(A^\sharp)_{ij} = a_{ji}^- \quad \forall i, j.$$

Note that as A has entries in $\mathbb{R}_{\max}$ and can be used to perform max-plus multiplication, $A^\sharp$ has entries in $\mathbb{R} \cup \{+\infty\}$ and can be used to perform min-plus multiplication.

Below we will use the following property of scalar conjugates: for $a, x \in \mathbb{R}_{\max}$ and $b \in \mathbb{R} \cup \{+\infty\}$, we have $a \otimes x \leq b$ if and only if $x \leq a^- \otimes' b$. This property can be easily extended to matrices:

Proposition 2.1 (e.g. [3, Theorem 1.6.25]): *Let $A \in \mathbb{R}_{\max}^{m \times n}, x \in \mathbb{R}_{\max}^n$ and $b \in (\mathbb{R} \cup \{+\infty\})^m$. Then*

$$A \otimes x \leq b \quad \text{if and only if} \quad x \leq A^\sharp \otimes' b.$$

The scalar and matrix conjugations discussed above were introduced by Cuninghame-Green [1] and as residuations of max-plus scalar and matrix multiplication by Baccelli et al. [2]. In particular, we use the notation $\sharp$ following Baccelli et al. [2] to emphasize the duality between the operator $A\otimes$ and its residuation $A^\sharp \otimes'$, but we prefer to use a more intuitive notation a^- and x^- for scalars and vectors.

2.3. Two-sided systems and min–max functions

The following is an obvious corollary of Proposition 2.1:

$$A \otimes x \le B \otimes x \quad \text{if and only if } x \le A^\sharp \otimes' (B \otimes x).$$

Here we assume that there is a finite entry in each row of B and each column of A. With this assumption, the inequality $x \le A^\sharp \otimes' (B \otimes x)$ can be written as the following system of inequalities:

$$x_j \le \min_{k:\, a_{kj}\in\mathbb{R}} \left(-a_{kj} + \max_{l:\, b_{kl}\in\mathbb{R}} (b_{kl} + x_l) \right) \quad \forall j = 1, .., n. \tag{1}$$

The expression on the right-hand side of (1) can be written as the component of a *min-max function* $f(x) = A^\sharp \otimes' (B \otimes x)$:

$$f_j(x) = \min_{k:\, a_{kj}\in\mathbb{R}} \left(-a_{kj} + \max_{l:\, b_{kl}\in\mathbb{R}} (b_{kl} + x_l) \right).$$

This function belongs to the class of topical functions investigated by Gaubert and Gunawardena [18]. For the present paper, we will only need that the *cycle-time vector* of f:

$$\chi(f) = \lim_{k\to\infty} \frac{f^k(0)}{k}, \tag{2}$$

exists and is well-defined [19]. Here, 0 denotes the vector with all components equal to 0.

In what follows, expression of the form $A^\sharp B$ will denote the min-max function $x \mapsto A^\sharp \otimes' (B \otimes x)$. In particular, note that, by default, the min-plus multiplication is always to be expected after the $\sharp$ sign. Cycle-time vectors of such functions $\chi(A^\sharp B)$ and their individual components $\chi_j(A^\sharp B)$ play a crucial role in the mean-payoff game approach to tropical optimization problems.

2.4. Mean payoff games

Consider the following zero-sum two player sequential game defined by two $m \times n$ matrices $A = (a_{ij})$ and $B = (b_{kl})$ over $\mathbb{R}_{\max}$. The game is played on the weighted directed graph (V, E, w) where the set of nodes $V = [m] \cup [n]$ is the union of

m nodes corresponding to the rows of the system $A \otimes x \leq B \otimes x$ and n nodes corresponding to the variables. The arc set E contains 1) the arcs (k, l) for which $b_{kl} \in \mathbb{R}$ and 2) the arcs (j, i) for which $a_{ij} \in \mathbb{R}$, and these arcs are weighted by $w(k, l) = b_{kl}$ and $w(j, i) = -a_{ij}$, respectively. The nodes in $[m]$ are the nodes at which player Max is active, and the nodes in $[n]$ are the nodes at which player Min is active. The game starts at a node $j \in [n]$ of Min, and first Min chooses to move a pawn to some node $i \in [m]$ of Max, via a weighted arc for which $a_{ij} \neq -\infty$. In doing so, player Max receives the payment $-a_{ij}$ from Min. Then player Max chooses to move to some $l \in [n]$ for which $b_{il} \neq -\infty$. Player Max then receives a payment of b_{il} from player Min, and then the game proceeds sequentially in turns. Player Max then wishes to maximize the average reward per turn of the infinite trajectory thus created, and Min wishes to minimize it.

Ehrenfeucht and Mycielski [20], who introduced this game, also showed that it is equivalent to a finite game, which ends as soon as the trajectory forms a cycle. In this finite game, we also assume that Max and Min play according to some positional strategies: they choose in advance some mapping $\sigma : [m] \to [n]$ and $\tau : [n] \to [m]$ such that Max always moves the pawn from $i \in [m]$ to $\sigma(i) \in [n]$ and Min moves the pawn from $j \in [n]$ to $\tau(j) \in [m]$, so that $b_{i\sigma(i)} \neq -\infty$ and $a_{\tau(j)j} \neq -\infty$ for all $i \in [m]$ and $j \in [n]$. The two positional strategies form a digraph, whose arc set is a subset of the edges in the original mean payoff game, with each node having a unique outgoing arc, either $(i, \sigma(i))$ or $(j, \tau(j))$ (see, for example, [14]). The mean weight per turn of the first cycle formed by the trajectory is then the reward of player Max, and the game is called winning for Max if this reward is nonnegative. Max then wants to maximize his reward by choosing a better positional strategy, and Min counterplays to minimize her loss by choosing her positional strategy.

Note that the assumption written above guarantees that Max and Min have at least one positional strategy that they can use.

The mathematical expression for the reward of Max in the finite mean-payoff game starting at node j of Min, given that Max employs positional strategy σ and Min employs positional strategy τ, is

$$\Phi_{A,B}(j, \tau, \sigma) = \frac{1}{k} \left(\sum_{t=1}^{k} \left(b_{r_t s_t} - a_{r_{t+1} s_t} \right) \right),$$

Here $r_1 s_1 r_2 s_2 \ldots r_k s_k r_1$ is the unique cycle formed by the trajectory, which starts at a node j of Min and develops according to the positional strategies σ and τ. It is assumed that $r_{k+1} = r_1$.

Ehrenfeucht and Mycielski [20] showed that such mean-payoff game (both the infinite and the finite version of it) has value in pure strategies. The following can be deduced from [20] and Gaubert and Gunawardena [19].

Proposition 2.2 ([19,20]): *For a mean payoff game, where $A, B \in \mathbb{R}_{\max}^{m \times n}$ both have a finite entry in each row and column, there exist τ^*, σ^* such that for all*

$j = 1, \ldots, n$

$$\min_{\tau \in \mathcal{T}} \Phi_{A,B}(j, \tau, \sigma^*) = \Phi_{A,B}(j, \tau^*, \sigma^*) = \max_{\sigma \in \mathcal{S}} \Phi_{A,B}(j, \tau^*, \sigma), \qquad (3)$$

where $\mathcal{T}$ and $\mathcal{S}$ denote, respectively, the sets of positional strategies available to Min and to Max. Moreover, $\Phi_{A,B}(j, \tau^*, \sigma^*) = \chi_j(A^\sharp B)$ for all j (where $\chi_j(A^\sharp B)$ is the jth component of the cycle-time vector defined in (2)).

Finally, the relationship between mean-payoff games and the solvability of tropical two-sided systems can be summarized in the following result.

Proposition 2.3 ([13, Theorem 3.1]): *For a mean payoff game, where $A, B \in \mathbb{R}_{\max}^{m \times n}$ both have a finite entry in each row and column and σ^* and τ^* are equilibrium strategies, we have $\Phi_{A,B}(j, \sigma^*, \tau^*)(= \chi_j(A^\sharp B)) \geq 0$ if and only if there exists $x \in \mathbb{R}_{\max}^n$ such that $A \otimes x \leq B \otimes x$ and $x_j > -\infty$.*

Let us now consider the following example of a mean payoff game.

Example 2.4: Let the matrices A, B, given by:

$$A = \begin{bmatrix} 3 & -\infty \\ 7 & -\infty \\ -\infty & 0 \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} 2 & -\infty \\ -\infty & 1 \\ -3 & 4 \end{bmatrix}.$$

The two-sided system $A \otimes x \leq B \otimes x$ is given by the following system of inequalities:

$$\begin{aligned} 3 + x_1 &\leq 2 + x_1 \\ 7 + x_1 &\leq 1 + x_2 \qquad\qquad (4) \\ x_2 &\leq \max(-3 + x_1, 4 + x_2). \end{aligned}$$

We immediately see from the first inequality that there is no solution with $x_1 > -\infty$. In fact, the solution set of this system is $\{(-\infty, t) \colon t \in \mathbb{R}_{\max}\}$.

The corresponding game is given in Figure 1 (left). Here squares denote the nodes at which Max makes a move and circles denote the nodes at which Min makes a move.

Let players Max and Min choose positional strategies $\sigma \colon 1 \to 1$, $2 \to 2$, $3 \to 2$ and $\tau \colon 1 \to 1$, $2 \to 3$, respectively. The game is then played on a subgraph of the graph of the mean-payoff game. This subgraph is shown in Figure 1 (right).

Clearly, the rewards of Max for the trajectories starting at the two nodes of Min are:

$$\Phi_{A,B}(1, \sigma, \tau) = -1, \quad \Phi_{A,B}(2, \sigma, \tau) = 4.$$

Also, it can be checked that $\sigma^* = \sigma$ and $\tau^* = \tau$ is an equilibrium pair of strategies of Max and Min, in the sense of the saddle point property (3). The above rewards
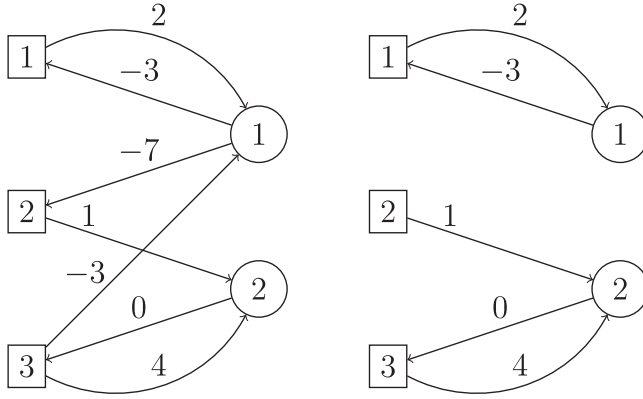
Figure 1. The mean payoff game corresponding to system (4) and its restriction.

are the values of the games starting at nodes 1 and 2 of Min. Here, Max is winning if the game starts at node 2 of Min but losing if the game starts at node 1 of Min. This correlates with the fact that system (4) has a solution with $x_2 > -\infty$, but not with $x_1 > -\infty$, as predicted by Proposition 2.3.

In what follows, we will often abbreviate mean payoff game(s) to MPG.

3. MPG representation of pseudolinear optimization

3.1. Pseudolinear optimization over alcoved polyhedra

Consider the following problem.

Problem 3.1 (Pseudolinear optimization over alcoved polyhedra): Given $U \in \mathbb{R}_{\max}^{n \times n}$, $p, g \in \mathbb{R}_{\max}^n$, $q, h \in (\mathbb{R} \cup \{+\infty\})^n$ such that $\rho(U) \leq 0$ and $U \otimes g \leq h$, find

$$
\begin{aligned}
\min_{x \in \mathbb{R}^n} \quad & x^- \otimes p \oplus q^- \otimes x \\
\text{subject to} \quad & g \leq x \leq h \\
& U \otimes x \leq x.
\end{aligned}
$$

and describe all $x \in \mathbb{R}^n$ for which the minimum is attained.

Note that whenever we have $q_i = +\infty$ it means that $q_i^- = -\infty$, so that x_i does not appear in the objective function (but x_i^- still may appear when $p_i \neq -\infty$). When $h_i = +\infty$, it means that x_i is not bounded from above by any constant, and if $g_i = -\infty$ then x_i is not bounded from below.

Problem 3.1 was considered and solved by Krivulin [7], for the case of real q and h. The solution set to the system of constraints in this problem is an alcoved polyhedron. The geometry of such polyhedra is combining tropical and ordinary convexity: see, e.g. Joswig and Kulas [21], or De La Puente and Claveria [22] for a more recent reference.

It follows from the results of [7] that an alcoved polyhedron described by

$$P = \{x \colon g \le x \le h,\ U \otimes x \le x\}$$

is non-empty if and only if the conditions $\rho(U) \le 0$ and $U^* \otimes g \le h$ hold. Note that these are precisely the conditions that are assumed in Problem 3.1.

The proof of the proposition below is due to Krivulin [7], although the case where some entries of h and q are equal to $+\infty$ was not considered in that work. See also Appendix of the present paper for a new alternative proof, which makes use of the connection to mean-payoff games.

Proposition 3.2 ([7, Theorem 6]): *The optimal value of Problem 3.1 is*

$$\theta = (q^- \otimes U^* \otimes p)^{\otimes \frac{1}{2}} \oplus h^- \otimes U^* \otimes p \oplus q^- \otimes U^* \otimes g,$$

If θ is finite then the solution set of this problem is

$$\{U^* \otimes v \colon g \oplus \theta^- \otimes p \le v \le (U^*)^{\#} \otimes' (\theta \otimes q \oplus' h),\ v \in \mathbb{R}^n\}. \tag{5}$$

Here and below, $t^{\otimes 1/2}$ is the same as $t/2$ for any $t \in \mathbb{R}_{\max}$ in the usual notation. (Note that one can define $t^{\otimes \alpha} := \alpha \times t$ for arbitrary $t \in \mathbb{R}_{\max}$ and $\alpha \ge 0$, but we will not need it in this paper.)

3.2. Pseudolinear optimization as parametric MPG

The purpose of this section is to recast Problem 1.1 as a parametric mean-payoff game. To begin, we can rewrite that problem in the following way, introducing new variable λ:

$$\begin{aligned} \min_{x \in \mathbb{R}^n, \lambda \in \mathbb{R}} \quad & \lambda \\ \text{subject to} \quad & x^- \otimes p \oplus q^- \otimes x \le \lambda, \\ & U \otimes x \oplus b \le V \otimes x \oplus d. \end{aligned} \tag{6}$$

The first inequality is equivalent to $x^- \otimes p \le \lambda$ and $q^- \otimes x \le \lambda$. The first of these inequalities is equivalent to $x_i^- \otimes p_i \le \lambda$ for all i, which is the same as $p_i \le x_i \otimes \lambda$ for all i. Hence we obtain that (6) is equivalent to:

$$\begin{aligned} \min_{x \in \mathbb{R}^n, \lambda \in \mathbb{R}} \quad & \lambda \\ \text{subject to} \quad & p \le \lambda \otimes x, \quad q^- \otimes x \le \lambda, \\ & U \otimes x \oplus b \le V \otimes x \oplus d. \end{aligned} \tag{7}$$

Introducing $z = (y\,t)^T$ where $y \in \mathbb{R}^n_{\max}$ and $t \in \mathbb{R}_{\max}$, we see that a finite solution to the system of constraints in (7) exists if and only if the following parametric

two-sided system

$$A \otimes z \leq B(\lambda) \otimes z,$$

where

$$A = \begin{pmatrix} U & b \\ -\infty & p \\ q^- & -\infty \end{pmatrix} \quad \text{and} \quad B(\lambda) = \begin{pmatrix} V & d \\ \lambda \otimes I & -\infty \\ -\infty & \lambda \end{pmatrix}, \qquad (8)$$

has a finite solution ($z \in \mathbb{R}^{n+1}$).

Thus, we reformulate Problem 1.1 as

$$\min\{\lambda \in \mathbb{R} \colon A \otimes z \leq B(\lambda) \otimes z \text{ is solvable with } z \in \mathbb{R}^{n+1}\},$$

where A and $B(\lambda)$ are given by (8).

We now reformulate this problem in terms of MPG, using Proposition 2.3, by introducing the function $\Phi(\lambda)$ defined as the minimal value of the MPG associated with the system $A \otimes z \leq B(\lambda) \otimes z$:

$$\min\{\lambda \in \mathbb{R} \colon \Phi(\lambda) \geq 0\}, \quad \text{where } \Phi(\lambda) = \min_i \chi_i(A^\# B(\lambda)).$$

We also define functions $\Phi_\tau(\lambda)$ and $\Phi^\sigma(\lambda)$ corresponding to MPG where the strategies of Min and Max are restricted to τ and σ, respectively:

$$\Phi_\tau(\lambda) = \min_i \chi_i(A_\tau^\# B(\lambda)), \quad \Phi^\sigma(\lambda) = \min_i \chi_i(A^\# B^\sigma(\lambda))$$

where

$$(A_\tau)_{ij} = \begin{cases} a_{ij} & \text{if } i = \tau(j) \\ -\infty & \text{otherwise.} \end{cases}, \quad (B^\sigma)_{ij} = \begin{cases} b_{ij} & \text{if } j = \sigma(i) \\ -\infty & \text{otherwise.} \end{cases}$$

Proposition 2.2 then implies the following result:

Proposition 3.3: *Let Σ be the set of all strategies of player Min and T be the set of all strategies of player Max. Then*

$$\min_{\tau \in T} \Phi_\tau(\lambda) = \Phi(\lambda) = \max_{\sigma \in \Sigma} \Phi^\sigma(\lambda). \qquad (9)$$

The following elementary properties of $\Phi(\lambda)$, $\Phi^\sigma(\lambda)$ and $\Phi_\tau(\lambda)$ are the same as in [14, Theorem 8].

Proposition 3.4: *For A and $B(\lambda)$ given by (8), functions $\Phi(\lambda)$, $\Phi^\sigma(\lambda)$ and $\Phi_\tau(\lambda)$ are non-decreasing, piecewise-linear and continuous.*
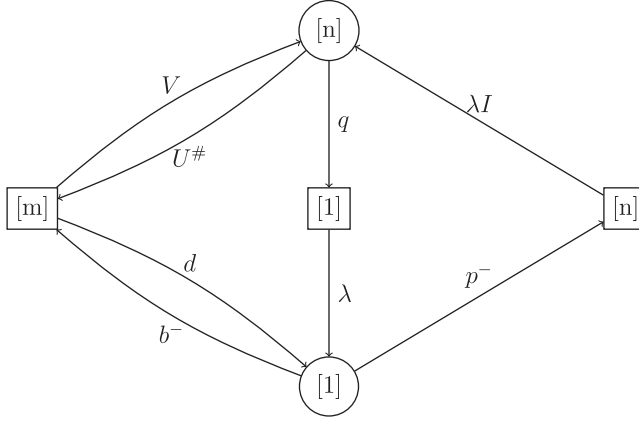
Figure 2. MPG corresponding to tropical pseudolinear programming.

3.3. MPG diagram of the problem

Analysing A and $B(\lambda)$ of (8) we can build an MPG diagram corresponding to this two-sided system, as shown in Figure 2. On this diagram we can see three groups of square nodes of Max: $[m]$, $[n]$ and $[1]$, with groups of arcs coming in and out of them, corresponding to $U \otimes x \oplus b \leq V \otimes x \oplus d$, $p \leq \lambda \otimes x$ and $q^- \otimes x \leq \lambda$, respectively (where m, n and 1 indicate the numbers of nodes in each group). Min has just two groups of circle nodes: $[n]$ and $[1]$, corresponding to the variables and to the free-standing column, respectively. An arc between two nodes of any two groups of nodes exists if and only if the corresponding entry of the matrix or the vector, by which the group connection is marked, is finite.

Using this MPG diagram we can establish the following optimality and unboundedness certificates for Problem 1.1, similar to Theorems 12 and 13 in [14]. Both certificates refer to some groups of nodes of Max shown in Figure 2. Their proofs, being similar to those in [14], are deferred to Appendix.

Proposition 3.5 (Optimality certificate): *λ^* is optimal if and only if $\Phi(\lambda^*) \geq 0$ and there exists τ such that in the mean-payoff game defined by A_τ and $B(\lambda^*)$ all cycles accessible from some node of Min have non-positive weights, and all cycles of zero weight accessible from that node of Min contain a node of Max that does not belong to the left group $[m]$ of its nodes.*

Proposition 3.6 (Unboundedness certificate): *The problem is unbounded if and only if, for some σ, all cycles of the graph defined by A and $B^\sigma(0)$ contain nodes of Max that are only from the $[m]$ group (on the left) and have a nonnegative weight.*

We now consider the case when all data in the problems are integer.

Proposition 3.7: *When the finite entries of p, q, U, b, V, d are all integer, the optimal value of Problem 1.1, if finite, is an integer multiple of 1/2.*

Proof: For the MPG in Figure 2, for each λ there is $\epsilon > 0$ such that

$$\Phi(\mu) = \Phi_{A,B(\mu)}(j,\sigma,\tau)$$

for some j, τ and σ and all $\mu \in [\lambda - \epsilon, \lambda]$. This implies that

$$\lambda^* = \min\{\lambda \colon \Phi_{A,B(\lambda)}(j,\sigma,\tau) \geq 0\} = \min\{\lambda \colon k\lambda + s \geq 0\},$$

for some j, σ, τ, k and s. Here we recall that $\Phi_{A,B(\lambda)}(j,\sigma,\tau)$ is the weight of a cycle (divided by the number of turns). Inspecting Figure 2 we can see that a cycle can collect no more than two repetitions of λ, hence in the above expression for λ^* we can have k equal to 1 or 2, which means that the denominator of the optimal value is bounded by 2. ∎

4. Bisection method

Since $\Phi(\lambda)$ is a non-decreasing function, we can use the bisection method to find $\min\{\lambda \colon \Phi(\lambda) \geq 0\}$. Recall that $\Phi(\lambda) \geq 0$ is equivalent to existence of $x \in \mathbb{R}^n$ that solves

$$
\begin{aligned}
x^- \otimes p \oplus q^- \otimes x &\leq \lambda, \\
U \otimes x \oplus b &\leq V \otimes x \oplus d.
\end{aligned}
\tag{10}
$$

In general, this method can be only approximate, but in the case of integer input (i.e. when all finite entries of U, V, b, d, p, q are integer) it can be made precise, since by Proposition 3.7 in this case the optimal λ is an integer multiple of 1/2. In the description of the bisection method given below, rounding up ($\lceil \cdot \rceil$) means finding the least λ greater than or equal to the given number and an integer multiple of 1/2. Similarly, rounding down ($\lfloor \cdot \rfloor$) means finding the biggest λ less than or equal to the given number and an integer multiple of 1/2.

Before we give a description of the bisection method, we need to have the upper and the lower bounds, with which we can start.

Upper bound: Using an MPG solver such as described in [23] or [24], find $x \in \mathbb{R}^n$ such that $U \otimes x \oplus b \leq V \otimes x \oplus d$. Then compute

$$\lambda_1^{(+)} = \left\lfloor x^- \otimes p \oplus q^- \otimes x \right\rfloor \tag{11}$$

for this x. Then we know that $\Phi(\lambda_1^{(+)}) \geq 0$.

Lower bound: We define $\lambda_1^{(-)}$ as the optimal value of the unconstrained problem

$$\min_{x \in \mathbb{R}^n} x^- \otimes p \oplus q^- \otimes x.$$

Using the result of Krivulin [6, Theorem 4] (a slight extension to the case $q \in (\mathbb{R} \cup \{+\infty\})^n$), we have

$$\lambda_1^{(-)} = (q^- \otimes p)^{\otimes \frac{1}{2}} \tag{12}$$

As this unconstrained problem is a relaxation of the problem in question, we either have that $\Phi(\lambda_1^{(-)}) \geq 0$ and then $\lambda_1^{(-)}$ is the optimal value of the constrained

problem, or we have $\Phi(\lambda_1^{(-)}) < 0$ and then we continue with the bisection method. If $q^- \otimes p$ is finite, which we assume, then the problem is bounded from below. Note that $q^- \otimes p$ is finite if and only if there exists at least one i such that both q_i and p_i are finite. We now describe the bisection algorithm.

Algorithm 1 Integer Bisection

Input: U, V, b, d, p with integer or $-\infty$ entries, q with integer or $+\infty$ entries.

1: Compute $\lambda_1^{(-)}$ by (12).
2: **if** $\Phi(\lambda_1^{(-)}) \geq 0$ **then**
3:      Find a finite solution x to (10) with $\lambda = \lambda_1^{(-)}$ and return.
4: **else**
5:      Compute $\lambda_1^{(+)}$ by (11) and proceed with $k = 1$.
6: **end if**
7: **while** $\lambda_k^{(-)} < \lambda_k^{(+)}$ **do**
8:      $\lambda_k = (\lambda_k^{(-)} + \lambda_k^{(+)})/2.$
9:      **if** $\Phi(\lambda_k) < 0$ **then**
10:          $\lambda_{k+1}^{(+)} = \lambda_k^{(+)}$ and $\lambda_{k+1}^{(-)} = \lceil \lambda_k \rceil.$
11:      **else**
12:          $\lambda_{k+1}^{(+)} = \lfloor \lambda_k \rfloor$ and $\lambda_{k+1}^{(-)} = \lambda_k^{(-)}.$
13:      **end if**
14:      $k = k + 1.$
15: **end while**
16: Find a finite solution x to (10) with $\lambda = \lambda_k^{(+)} = \lambda_k^{(-)}$ and return.

Output: (λ, x).

It is clear that in this algorithm, for each $\lambda_k^{(+)}$ we have $\Phi(\lambda_k^{(+)}) \geq 0$, and for each $\lambda_k^{(-)}$ we have $\Phi(\lambda_k^{(-)}) \leq 0$ and $\Phi(\lambda') < 0$ whenever $\lambda' < \lambda_k^{(-)}$. This implies that when $\lambda = \lambda_k^{(+)} = \lambda_k^{(-)}$, we indeed have $\Phi(\lambda) = 0$ and $\Phi(\lambda') < 0$ for any $\lambda' < \lambda$, so this λ is the optimal value of the problem.

Example 4.1: This example demonstrates that $(q^- \otimes p)^{\otimes 1/2}$ does not always work as a good lower bound. Consider a pseudolinear optimization problem (Problem 1.1) with

$$U = \begin{pmatrix} 0 & -\infty \\ -\infty & 0 \end{pmatrix}, \quad b = \begin{pmatrix} -\infty \\ -\infty \end{pmatrix}, \quad V = \begin{pmatrix} -\infty & 0 \\ 0 & -\infty \end{pmatrix},$$

$$d = \begin{pmatrix} -\infty \\ -\infty \end{pmatrix},$$

$$p = \begin{pmatrix} 0 \\ -\infty \end{pmatrix}, \quad q = \begin{pmatrix} +\infty \\ 0 \end{pmatrix}.$$

In this example, p_i and q_i are not finite at the same time for any i, so $(q^- \otimes p)^{\otimes 1/2} = -\infty$, but the problem is bounded and its optimal value is 0. Indeed, we are seeking minimum over $\max(x_1^-, x_2)$ on the line $x_1 = x_2$, which is equal to 0.

5. Newton iterations

5.1. Formulation

Let us begin this section by introducing the concepts of left-optimal strategy. By the right-hand side of (9), $\Phi(\cdot)$ is a pointwise maximum of a finite number of functions $\Phi^\sigma(\cdot)$, for $\sigma \in \Sigma$. Therefore, for each λ there exists $\Sigma' \subseteq \Sigma$ such that $\Phi^\sigma(\lambda) = \Phi(\lambda)$ for each $\sigma \in \Sigma'$. Furthermore, each of these $\Phi^\sigma(\cdot)$ is a piecewise-linear and continuous function, so there exists a small enough $\epsilon > 0$ such that each $\Phi^\sigma(\cdot)$ with $\sigma \in \Sigma'$ is linear on $[\lambda - \epsilon, \lambda]$ and is bigger than any $\Phi^\sigma(\cdot)$ with $\sigma \notin \Sigma'$. This implies that there exists σ^* such that $\Phi^{\sigma^*}(\mu) = \Phi(\mu)$ for all $\mu \in [\lambda - \epsilon, \lambda]$. Such σ^* is called a *left-optimal strategy* at λ. Left-optimal strategies play the role of derivatives in Algorithm 2 stated below. Iterations of this algorithm will refer to efficient solution of the following problem

$$\min\{\lambda : \Phi^\sigma(\lambda) \geq 0\}, \tag{13}$$

which will be explained in Section 5.3.

Algorithm 2 Newton iterations, left-optimal strategies

Input: U, V, b, d, p with entries in $\mathbb{R}_{\max}$ and q with entries in $\mathbb{R} \cup \{+\infty\}$.
  1: Set $\lambda_0 = +\infty$, compute λ_1 by (11) and proceed with $k = 1$.
  2: **while** $\lambda_k < \lambda_{k-1}$ **do**
  3:    Find a left-optimal strategy σ_k of Player Max at λ_k.
  4:    Solve (13) with $\sigma = \sigma_k$ and let λ_{k+1} be the optimal value of (13).
  5:    $k = k + 1$.
  6: **end while**
  7: Find a solution $x \in \mathbb{R}^n$ to system (10) with $\lambda = \lambda_k$.
Output: (λ, x).

The following result and its proof are similar to the corresponding statement from Gaubert et al. [14], therefore the proof is omitted.

Proposition 5.1: *Newton algorithm finds a solution of a tropical pseudoquadratic optimization problem in a finite number of steps, limited by the number of strategies of player Max in the associated MPG.*

In the general case (i.e. when the data are arbitrary real numbers) left-optimal strategies can be found using the algebra of germs, for which we also refer the reader to Gaubert et al. [14].

5.2. The case of integer data

In this section we will discuss how to implement Newton's algorithm in the case when all given data (i.e. coefficients of U, V, b, d, p and q) are integer.

We first explain how to find the left-optimal strategies in the case of integer data. By the arguments similar to those of Proposition 3.7, λ_k appearing in Newton iterations are of the form $l/2$ where l is an integer, while the breakpoints of $\Phi(\lambda)$ can be at rational points with denominators not exceeding $2(n + 1)$. The greatest denominator of a distance between λ_k and such a breakpoint is bounded from above by $4(n + 1)$. Therefore, if we take $\epsilon = 1/4(n + 1)$, we can be sure that if σ_k is optimal at $\lambda_k - \epsilon$ then it is optimal for any $\lambda \in [\lambda_k - \epsilon, \lambda_k]$, thus left-optimal.

However, in the case of integer data, instead of finding a left-optimal strategy we can find an optimal strategy σ_k at $\lambda_k^{[-]}$, defined as the largest multiple of $1/2$ less than λ_k if it is not a multiple of $1/2$ (which may happen if $k = 1$), and as $\lambda_k - 1/2$ if it is. We use the notation $\lambda_k^{[-]}$ here to distinguish it from the notation $\lambda_k^{(-)}$ (and $\lambda_k^{(+)}$) used in the bisection method.

If we have $\Phi^{\sigma_k}(\lambda_k^{[-]}) \geq 0$, then the Newton iterations proceed, and if $\Phi^{\sigma_k}(\lambda_k^{[-]}) < 0$ then $\Phi(\lambda) < 0$ for all $\lambda < \lambda_k$ and we must stop. So we have the following modification of Algorithm 2, where optimal strategies can be found using some MPG solvers such as the algorithm by Dhingra and Gaubert [23].

Algorithm 3 Integer Newton iterations

Input: U, V, b, d, p with integer or $-\infty$ entries, and q with integer or $+\infty$ entries.

1: Set $\lambda_0 = +\infty$, compute λ_1 by (11) and proceed with $k = 1$.
2: **while** $\lambda_k < \lambda_{k-1}$ **do**
3:     Compute $\lambda_k^{[-]}$ and find an optimal strategy σ_k of player Max at $\lambda_k^{[-]}$.
4:     Solve (13) with $\sigma = \sigma_k$ and let λ_{k+1} be the optimal value of (13).
5:     $k = k + 1$.
6: **end while**
7: Find a finite solution x to system (10) with $\lambda = \lambda_{k-1}$.

Output: (λ, x).

Note that in this algorithm the sequence of λ_i is strictly decreasing until the last step, at which we may have $\lambda_k = \lambda_{k-1}$, $\lambda_k > \lambda_{k-1}$ or even $\lambda_k = +\infty$ if the problem (13) is infeasible for $\sigma = \sigma_k$.

5.3. Finding the least zero of a partial spectral function

Here we discuss how to solve (13). The problem can be translated back to two-sided system where it becomes the problem of finding

$$\min\{\lambda \in \mathbb{R} \colon A \otimes z \le B^{\sigma}(\lambda) \otimes z \text{ has a solution } z \in \mathbb{R}^{n+1}\}, \qquad (14)$$

where B^{σ} is the matrix with entries equal to $b_{ij}^{\sigma} = b_{ij}$ when $j = \sigma(i)$ and to $-\infty$ otherwise, and σ is the strategy of Max appearing in (13). Recalling (8), we can rewrite (14) as

$$\min_{\lambda \in \mathbb{R},\, x \in \mathbb{R}^n} \lambda \quad \text{s.t. } p \le \lambda \otimes x, \quad q^{-} \otimes x \le \lambda, \quad U \otimes x \oplus b \le V^{\sigma} \otimes x \oplus d^{\sigma}. \tag{15}$$

where $(V^{\sigma}\ d^{\sigma}) = (V\ d)^{\sigma}$, and $(V\ d)^{\sigma}$ is defined as B^{σ} above: we leave the entries of B for which $j = \sigma(i)$ untouched and set all the remaining entries to $-\infty$. We now work with the system of constraints

$$U \otimes x \oplus b \le V^{\sigma} \otimes x \oplus d^{\sigma}. \tag{16}$$

Proposition 5.2: *System* (16) *is equivalent to*

$$F \otimes x_J \oplus G \otimes x_{\bar{J}} \le x_J, \quad l_J \le x_J \le u_J \quad x_{\bar{J}} \le u_{\bar{J}}. \tag{17}$$

and further to

$$R \otimes x \le x, \quad l \le x \le u \quad \text{for } R = \begin{pmatrix} F & G \\ -\infty & -\infty \end{pmatrix}$$

for some set $J \subseteq [n]$, its complement $\bar{J} = [n] \setminus J$, matrices F, G, and vectors l and u.

Proof: Define the index set J as follows:

$$J = \{j \in [n] : \sigma(i) = j \text{ for some } i\}.$$

Now for any $j \in J$ consider all i such that $\sigma(i) = j$. Denote the set of such i by I_j and observe that $\bigcup_{j \in J \cup \{n+1\}} I_j = [m]$. We have two cases:
    Case 1: $j \in J$. In this case we obtain

$$\bigoplus_{k=1}^{n}(v_{ij}^{-} \otimes u_{ik} \otimes x_k) \oplus v_{ij}^{-} \otimes b_i \le x_j, \quad j \in J, \quad i \in I_j \tag{18}$$

for all such $i \in I_j$, and summing it up over $i \in I_j$ we have

$$\bigoplus_{k=1}^{n}\bigoplus_{i \in I_j}(v_{ij}^{-} \otimes u_{ik} \otimes x_k) \oplus \bigoplus_{i \in I_j}(v_{ij}^{-} \otimes b_i) \le x_j, \quad j \in J. \tag{19}$$

Observe that (19) is equivalent to the system of inequalities (18) where i runs over I_j, and equivalent to the subsystem of (16), consisting of inequalities i such that $\sigma(i) \in [n]$.

Case 2. $j = n + 1$. In this case we obtain

$$\bigoplus_{k=1}^{n} (u_{ik} \otimes x_k) \oplus b_i \le d_i, \quad i \in I_{n+1}$$

for all $i \in I_{n+1}$. Considering these inequalities for all $i \in I_{n+1}$ and seeing that $b_i \le d_i$ in this case is just a necessary condition for (16) to be consistent, we see that the system of such inequalities taken over $i \in I_{n+1}$ is equivalent to the system

$$x_k \le \min_{i \in I_{n+1}} u_{ik}^- \otimes' d_i. \tag{20}$$

This is a system of upper bounds on x_k (some of them can be equal to $+\infty$ if all the corresponding u_{ik} are $-\infty$).

Equation (16) is thus equivalent to the system of (19) and (20). Combining these two, one can easily recognize (17).

Next we observe that the constraint given by $F \otimes x_I \oplus G \otimes x_J \le x_I$ can be written as

$$\begin{bmatrix} F & G \\ -\infty & -\infty \end{bmatrix} \otimes \begin{bmatrix} x_I \\ x_J \end{bmatrix} \le \begin{bmatrix} x_I \\ x_J \end{bmatrix}$$

which is equivalent to $R \otimes x \le x$. ∎

The above proposition implies that the problem which we have to solve at each iteration is the following problem

$$\min_{x \in \mathbb{R}^n} x^- \otimes p \oplus q^- \otimes x \quad \text{s.t.} \ l \le x \le u \quad \text{and} \quad R \otimes x \le x. \tag{21}$$

Recognizing Problem 3.1, we recall that Proposition 3.2 yields the optimal value

$$\theta = (q^- \otimes R^* \otimes p)^{\otimes \frac{1}{2}} \oplus u^- \otimes R^* \otimes p \oplus q^- \otimes R^* \otimes l \tag{22}$$

and the solution set

$$x = \{R^* \otimes v : l \oplus \theta^- \otimes p \le v \le (R^*)^\# \otimes' (\theta \otimes q \oplus' u), \ v \in \mathbb{R}^n\}. \tag{23}$$

of this problem. Thus we can compute $\lambda_k = \theta$ using (22) and at optimality we can take any vector from (23) as a solution of the system of constraints in (15) and hence also system (10) with $\lambda = \lambda_k = \theta$. For a finite vector v we can take a vector with the following components:

$$v_i = \begin{cases} ((R^*)^\# \otimes' (\theta \otimes q \oplus' u))_i, & \text{if } ((R^*)^\# \otimes' (\theta \otimes q \oplus' u))_i \ne +\infty, \\ l_i \oplus \theta^- \otimes p_i, & \text{otherwise if } l_i \oplus \theta^- \otimes p_i \ne -\infty, \\ 0, & \text{otherwise.} \end{cases}$$

Note that the computation of θ in (22) requires no more than $O(n^3)$ operations and can be performed very efficiently by shortest path algorithms.

6. Example and numerical experiments

6.1. Example

Consider the pseudolinear optimization problem where

$$U = \begin{pmatrix} -\infty & -2 \\ 3 & -\infty \end{pmatrix}, \quad b = \begin{pmatrix} -\infty \\ -\infty \end{pmatrix},$$

$$V = \begin{pmatrix} 1 & 0 \\ -\infty & 1 \end{pmatrix}, \quad d = \begin{pmatrix} -\infty \\ 1 \end{pmatrix},$$

$$p = \begin{pmatrix} 0 \\ -\infty \end{pmatrix}, \quad q = \begin{pmatrix} -1 \\ 0 \end{pmatrix}. \tag{24}$$

6.1.1. Application of bisection (Algorithm 1)

Start: We first compute $\lambda_1^{(-)} = (q^- \otimes p)^{\otimes \frac{1}{2}} = \frac{1}{2}$. We find that $\Phi(\lambda_1^{(-)}) = -\frac{1}{3} < 0$, so we proceed. Using the alternating method of [24], we obtain a finite solution $x^0 = (-8 \ -8)^\top$ for $U \otimes x \oplus b \leq V \otimes x \oplus d$. Let $\lambda_1^{(+)} = (x^0)^- \otimes p \oplus q^- \otimes x^0 = 8$.

Iterations: For $k = 1$, let $\lambda_1 = \frac{\lambda_1^{(+)} + \lambda_1^{(-)}}{2} = \frac{17}{4}$. Since $\Phi(\lambda_1) = 2 > 0$, set $\lambda_2^{(+)} = \lfloor \lambda_1 \rfloor = 4$ and $\lambda_2^{(-)} = \lambda_1^{(-)} = \frac{1}{2}$.

For $k = 2$, we check that $\lambda_2^{(+)} \neq \lambda_2^{(-)}$ and compute $\lambda_2 = \frac{\lambda_2^{(+)} + \lambda_2^{(-)}}{2} = \frac{9}{4}$. Since $\Phi(\lambda_2) = \frac{5}{6} > 0$, set $\lambda_3^{(+)} = \lfloor \lambda_2 \rfloor = 2$ and $\lambda_3^{(-)} = \lambda_2^{(-)} = \frac{1}{2}$.

For $k = 3$, we check that $\lambda_3^{(+)} \neq \lambda_3^{(-)}$ and compute $\lambda_3 = \frac{\lambda_3^{(+)} + \lambda_3^{(-)}}{2} = \frac{5}{4}$. Since $\Phi(\lambda_3) = \frac{1}{6} > 0$, set $\lambda_4^{(+)} = \lfloor \lambda_3 \rfloor = 1$ and $\lambda_4^{(-)} = \lambda_3^{(-)} = \frac{1}{2}$.

For $k = 4$, we check that $\lambda_4^{(+)} \neq \lambda_4^{(-)}$ and compute $\lambda_4 = \frac{\lambda_4^{(+)} + \lambda_4^{(-)}}{2} = \frac{3}{4}$. Since $\Phi(\lambda_4) = -\frac{1}{6} < 0$, set $\lambda_5^{(+)} = \lambda_4^{(+)} = 1$ and $\lambda_5^{(-)} = \lceil \lambda_4 \rceil = 1$.

We have $\lambda_5^{(-)} = \lambda_5^{(+)} = 1$, so we stop, this is the optimal value of the problem. Using the alternating method, we obtain a finite solution $x = (-1 \ 1)^T$ for system (10) with $\lambda = \lambda_5^{(-)} = 1$. Then return $(\lambda_5^{(-)}, x)$ as a solution of the problem.

6.1.2. Application of Newton method (Algorithm 3)

Start. We begin with the same $\lambda_1 = 8$ as in the bisection method.

Iterations: We first find an optimal strategy σ_1 at $\lambda_1^{[-]} = 7.5$ with $\sigma_1(1) = 1$, $\sigma_1(2) = 3$, where 1 and 2 are nodes of Max that correspond to the inequalitites of the system $U \otimes x \oplus b \leq V \otimes x \oplus d$ (recall that Max does not have choice at any other node of the associated MPG).

Following Proposition 5.2, we find that $\min\{\lambda : \Phi^{\sigma_1}(\lambda) \geq 0\}$ is equivalent to (21), where

$$R = \begin{pmatrix} -\infty & -3 \\ -\infty & -\infty \end{pmatrix}, \quad l = \begin{pmatrix} -\infty \\ -\infty \end{pmatrix}, \quad u = \begin{pmatrix} -2 \\ +\infty \end{pmatrix}.$$

Then we obtain

$$\lambda_2 = (q^- \otimes R^* \otimes p)^{\otimes \frac{1}{2}} \oplus u^- \otimes R^* \otimes p \oplus q^- \otimes R^* \otimes l = 2.$$

At iteration 2 we check that $\lambda_2 < \lambda_1$ and find an optimal strategy σ_2 at $\lambda_2^{[-]} = 1.5$ with $\sigma_2(1) = 2$ and $\sigma_2(2) = 2$. Following Proposition 5.2, we next solve (21) with

$$R = \begin{pmatrix} -\infty & -\infty \\ 2 & -\infty \end{pmatrix}, \quad l = \begin{pmatrix} -\infty \\ -\infty \end{pmatrix}, \quad u = \begin{pmatrix} +\infty \\ +\infty \end{pmatrix}.$$

We obtain

$$\lambda_3 = (q^- \otimes R^* \otimes p)^{\otimes \frac{1}{2}} \oplus u^- \otimes R^* \otimes p \oplus q^- \otimes R^* \otimes l = 1.$$

At iteration 3, we check that $\lambda_3 < \lambda_2$ and find an optimal strategy σ_3 at $\lambda_3^{[-]} = 0.5$ with $\sigma_3(1) = 2$ and $\sigma_3(2) = 2$, which is the same as at the previous iteration. Obviously, we obtain $\lambda_4 = 1 = \lambda_3$, so we **stop**. As $\sigma_2 = \sigma_3$ is an optimal strategy also at $\lambda_3 = \lambda_4 = 1$, we can find a finite solution using

$$x = R^* \otimes ((R^*)^\sharp \otimes' (\theta \otimes q \oplus' u)) = (-1 \ 1)^\top$$

and return $\lambda = 1$ and $x = (-1 \ 1)^\top$ as an optimal solution to the problem. The Newton iterations are shown on Figure 3.
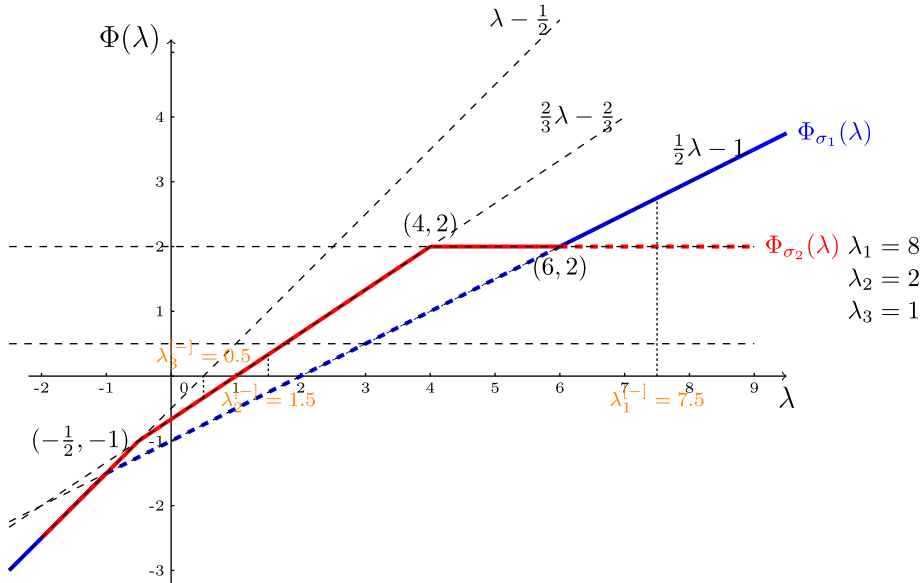

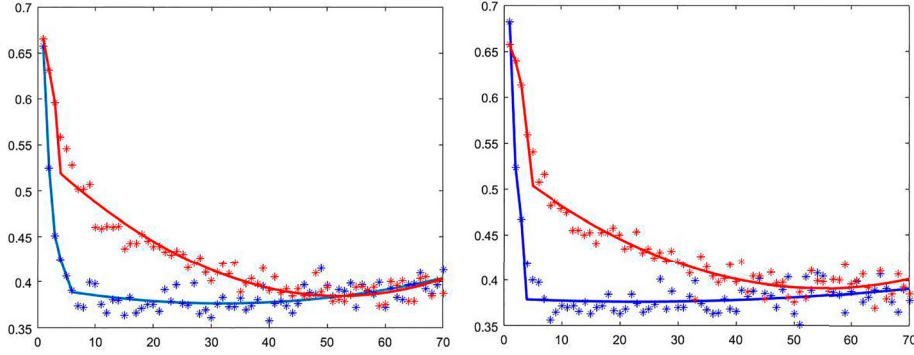**Figure 3.** Application of Newton method to the problem given by (24).

Figure 4. Percentage of the cases where $(q^- \otimes p)^{\otimes 1/2}$ is an optimal value for $m = n$ from 1 to 70. Finite entries are randomly picked from $[-500, 500]$ (left) and from $[-500000, 500000]$ (right). Results for the cases where all entries are finite (lower graphs) are shown versus the cases where approximately 30% of all entries are finite (upper graphs).

6.2. Numerical experiments

We implemented Algorithms 1 and 3 in MATLAB and ran some numerical experiments. Before running these experiments we checked the percentage of cases for which $(q^- \otimes p)^{\otimes 1/2}$ (the optimal value of the unconstrained problem) is an optimal value of the constrained problem so that no algorithm is required. Before checking this, the program checked that the system of constraints is feasible and $(q^- \otimes p)^{\otimes 1/2}$ is finite. For $m = n$ ranging from 1 to 70 we ran 2000 experiments for each dimension for the cases where the entries are all finite or we have approximately 30% of finite entries, and where the finite entries are randomly and uniformly selected in the interval $[-500, 500]$ or $[-500000, 500000]$ (for all four combinations). The results are shown in Figure 4. We see that, in general, for very low dimensions it is highly likely that $(q^- \otimes p)^{\otimes 1/2}$, but the probability of this quickly falls to the level of 40% or slightly below, and then starts to grow very slowly. We also observed how this percentage behaves for dimensions up to 400 performing just 20 experiments for each dimension and each situation, and recorded that $(q^- \otimes p)^{\otimes 1/2}$ was optimal in 41% of all solvable instances with finite $q^- \otimes p$.

We also performed similar experiments with rectangular matrices with 50 rows and number of columns ranging from 35 to 90, see Figure 5. Unlike in the previous experiments, the results do not depend on the range of entries or sparsity. The percentage of cases where $(q^- \otimes p)^{\otimes 1/2}$ is optimal increases monotonically from $15 - 20\%$ for $n = 35$ to almost 100% when $n = 90$. Cases $n < 35$ were not checked, because for such low n most of the randomly generated constraint systems become infeasible.

Next, we checked the performance of Newton and bisection algorithms, in the same vein as in Gaubert et al. [14]. In our experiments, U and V were square matrices with dimensions ranging from 1 to 400, with the range of finite entries
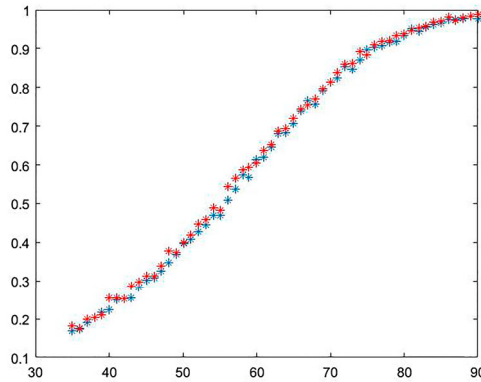
Figure 5. Percentage of the cases where $(q^- \otimes p)^{\otimes 1/2}$ is an optimal value for $m = 50$ and n from 35 to 90.

$[-500, 500]$ or $[-500000, 500000]$, with finite entries only or with approximately 30% of finite entries. Results are shown in Figures 6 and 7. For each dimension, both methods were run exactly once, after repeatedly ignoring the cases where the system of constraints was infeasible or the value $(q^- \otimes p)^{\otimes 1/2}$ was infinite or optimal for the generated problem. We see that, as in the case of linear-fractional programming [14], the 'average number' of Newton iterations (computed here for dimension d as the average taken over dimensions in the range $[d - 10, d + 10]$ and shown by thick broken line) slowly grows with dimension and, in the case of entries ranging in $[-500, 500]$ becomes similar to the number of bisection iterations before the dimension 400 is reached. Unlike in the case of linear-fractional programming, here we were able to implement bisection also in the case of sparse data (30% of finite entries). Similarly to [14], we observed that the number of bisection iterations grows with the range of the finite entries (for an obvious reason, since it means that the interval between lower and upper bounds increases). In particular, the gap between bisection and Newton iterations still remains quite big for $d = 400$ for $[-500000, 500000]$. However, the number of bisection iterations also decreases with the increase in dimension: this is different from what was observed in tropical linear-fractional programming [14] where the number of these iterations was stable.

7. Pseudoquadratic optimization

In this section we will consider pseudoquadratic optimization problem with two-sided constraints: Problem 1.2. Like Problem 1.1, which we considered earlier, it can be also represented by parametric MPG and solved by means of bisection and Newton methods, calling an MPG solver at each iteration. Below we discuss the details of it, as well as similarity and difference between the pseudoquadratic and pseudolinear tropical optimization.
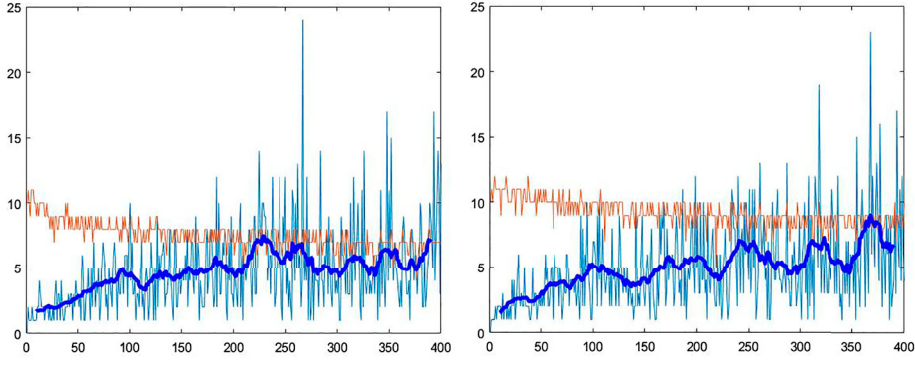
Figure 6. Number of iterations of bisection (upper graphs) and Newton iterations (lower graphs) for randomly generated problems of dimension 1 to 400 with the range of entries between -500 and 500 with finite entries only (left) and with approximately 30% of finite entries (right).
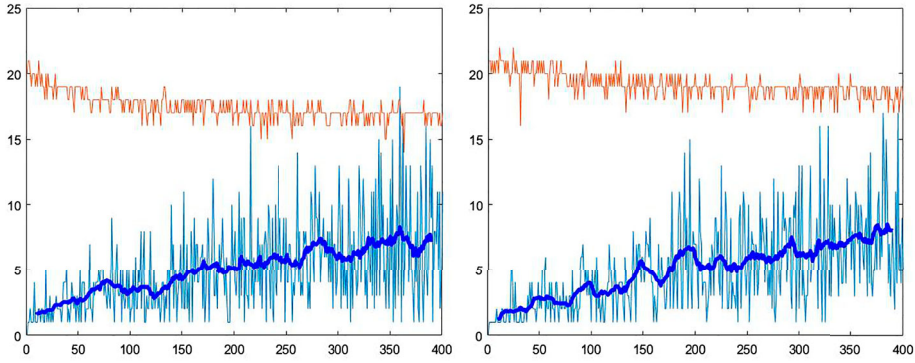

Figure 7. Number of iterations of bisection (upper graphs) and Newton iterations (lower graphs) for randomly generated problems of dimension 1 to 400 with the range of entries between -500000 and 500000 with finite entries only (left) and with approximately 30% of finite entries (right)

As a starter, using that $x^- \otimes C \otimes x \leq \lambda$ is equivalent to $C \otimes x \leq \lambda \otimes x$, we can recast Problem 1.2 as the problem of finding the least λ such that $A \otimes z \leq B(\lambda) \otimes z$, is solvable with $z \in \mathbb{R}^{n+1}$, where

$$A = \begin{pmatrix} U & b \\ C & -\infty \\ -\infty & p \\ q^- & -\infty \end{pmatrix} \quad \text{and} \quad B(\lambda) = \begin{pmatrix} V & d \\ \lambda \otimes I & -\infty \\ \lambda \otimes I & -\infty \\ -\infty & \lambda \end{pmatrix}. \tag{25}$$

The mean-payoff game diagram corresponding to this system is given in Figure 8. Comparing it with Figure 2, we see a new group of nodes $[n]$ on top of the diagram, corresponding to the inequalities $C \otimes x \leq \lambda \otimes x$. With respect to this game, we are solving the problem $\min\{\lambda \colon \Phi(\lambda) \geq 0\}$, where $\Phi(\lambda) = \min_i \chi_i(A^\sharp B(\lambda))$, with A and $B(\lambda)$ defined in (25). The theory of Section 3.2 applies verbatim.
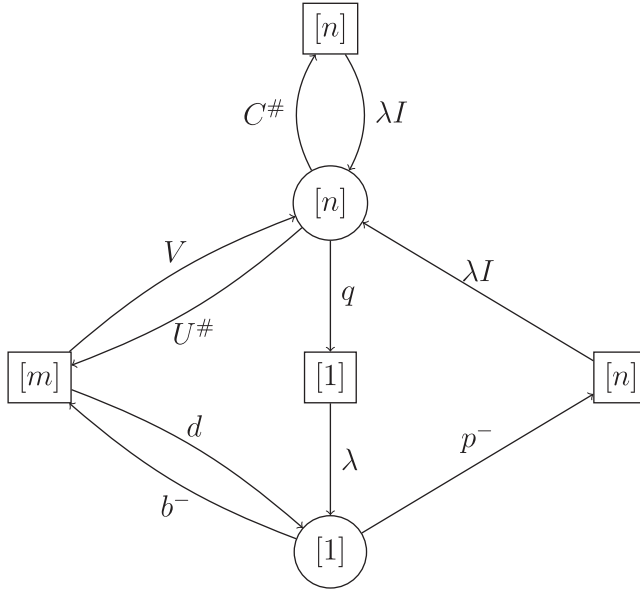
Figure 8. MPG diagram corresponding to pseudoquadratic optimization.

It can be also checked that both certificates stated in Propositions 3.5 and 3.6 extend to the pseudoquadratic optimization with no change, now referring to the groups of nodes in Figure 8. However, in the pseudoquadratic programming the cycles in the graph of the game can collect up to $n + 1$ repetitions of λ. Therefore, in the case of integer data, we can only say that the denominator of optimal value is bounded from above by $n + 1$. The same is true for any reduced MPG defined by A and $B^\sigma(\lambda)$.

An initial upper bound for the bisection method, following Section 4, can be computed as

$$\lambda_0^{(+)} = x^- \otimes C \otimes x \oplus q^- \otimes x \oplus x^- \otimes p,$$

where $x \in \mathbb{R}^n$ is a solution to $U \otimes x \oplus b \leq V \otimes x \oplus d$. The lower bound comes from the unconstrained pseudoquadratic problem $\min_{x \in \mathbb{R}^n} x^- \otimes C \otimes x \oplus q^- \otimes x \oplus x^- \otimes p$, the optimal value of which was found by Krivulin [10] to be

$$\lambda_0^{(-)} = \rho(C) \oplus (q^- \otimes p)^{\otimes 1/2}.$$

With these initial bounds we can run a usual bisection scheme as an approximate method, or we can still use Algorithm 1 as its exact version, but rounding up ($\lceil \cdot \rceil$) now means finding the least rational number greater than or equal to the given number and with denominator bounded by $n + 1$. Similarly, rounding down ($\lfloor \cdot \rfloor$) means finding the biggest rational number less than or equal to the given one and with denominator bounded by $n + 1$. These operations are still not too difficult: suppose that λ is not a integer and set $\lambda = a + \frac{c}{b}$, where $a \in \mathbb{Z}, b \in \mathbb{N}, c \in \mathbb{N} \cup \{0\}$

and $\frac{c}{b}$ is an irreducible proper fraction if $c \neq 0$. Then we have

$$\lceil \lambda \rceil = \begin{cases} \lambda, & \text{if } c = 0 \text{ or } b \leq n+1, \\ a + \frac{e_1}{d_1}, & \text{otherwise,} \end{cases} \qquad \lfloor \lambda \rfloor = \begin{cases} \lambda, & \text{if } c = 0 \text{ or } b \leq n+1, \\ a + \frac{e_2}{d_2}, & \text{otherwise,} \end{cases}$$

(26)

where

$$\frac{e_1}{d_1} = \min \left\{ \frac{\lceil \lceil \frac{cd+1}{b} \rceil \rceil}{d} : \; 1 \leq d \leq n+1 \right\},$$

and

$$\frac{e_2}{d_2} = \max \left\{ \frac{\lfloor \lfloor \frac{cd-1}{b} \rfloor \rfloor}{d} : \; 1 \leq d \leq n+1 \right\}, \qquad (27)$$

and $\lceil \lceil \cdot \rceil \rceil$ and $\lfloor \lfloor \cdot \rfloor \rfloor$, respectively, mean the usual operations of rounding up and rounding down to the nearest integers, respectively.

We also have Newton iterations (Algorithm 2) based on left-optimal strategies, which are found using the algebra of germs in general case. In the case of integer data, the denominators of λ_k do not exceed $n+1$ and the denominators of break-points of $\Phi(\lambda)$ do not exceed $(n+1)^2$. Hence the largest possible denominator of the distance between λ_k and a breakpoint does not exceed $(n+1)^3$, so we can take $\epsilon = 1/(n+1)^3$ to ensure that a strategy that is optimal at $\lambda - \epsilon$ is optimal for the whole interval $[\lambda - \epsilon, \lambda]$.

Integer version of Newton iterations (Algorithm 3) also works, but in this algorithm $\lambda^{[-]}$ should be defined as the largest rational number that is (1) strictly smaller than λ, (2) has a denominator bounded by $n+1$. Following the notation in (26), we can obtain:

$$\lambda^{[-]} = \begin{cases} \lambda - \dfrac{1}{n+1}, & \text{if } \lambda \text{ is integer,} \\ a + \dfrac{e_2}{d_2}, & \text{otherwise,} \end{cases}$$

where $\frac{e_2}{d_2}$ is defined as in (27).

Solution of (13) is an important ingredient in any modification of Newton algorithm. This is treated exactly as in Section 5.3, since $C \otimes x \leq \lambda \otimes x$ does not add to the available strategies of Max, and they are still determined by the right-hand side of the system $U \otimes x \oplus b \leq V \otimes x \oplus d$. Proposition 5.2 then leads us to a problem of the following type:

$$\min_{x \in \mathbb{R}^n} x^- \otimes p \oplus q^- \otimes x \oplus x^- \otimes C \otimes x$$

$$\text{s.t.} \quad l \leq x \leq u \quad \text{and} \quad R \otimes x \leq x.$$

An explicit expression for the optimal value of this problem and solution set was obtained in Krivulin [8, Theorem 4] and it can be used instead of (22) and (23). However, the computational complexity of computing the optimal value of this problem rises to $O(n^5)$.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

References

[1] Cuninghame-Green RA. Minimax algebra. Berlin: Springer-Verlag; 1979. (Lecture Notes in Economics and Mathematical Systems; 166.

[2] Baccelli F, Cohen G, Olsder GJ, et al. Synchronization and linearity: an algebra for discrete event systems. Chichester: John Wiley & Sons Ltd; 1992.

[3] Butkovič P. Max-linear systems: theory and algorithms. London: Springer Science & Business Media; 2010.

[4] Zimmermann K. Extremální algebra. Útvar vědeckých informací ekonomického ústavu ČSAV, Prague, 1976. in Czech.

[5] Zimmermann U. Linear and combinatorial optimization in ordered algebraic structures. Amsterdam: North Holland; 1981. (Annals of Discrete Mathematics; 10).

[6] Krivulin N. A new algebraic solution to multidimensional minimax location problems with Chebyshev distance. WSEAS Trans Math. 2012;11(7):605–614.

[7] Krivulin N. Complete solution of a constrained tropical optimization problem with application to location analysis. Relational and algebraic methods in computer science 2014. Springer; 2014. p. 362–378. (LNCS; 8428).

[8] Krivulin N. A constrained tropical optimization problem: complete solution and application example. In: Litivinov GL, Sergeev SN, editors. Tropical and idempotent mathematics and applications. Vol. 616. Providence (RI): AMS; 2014. p. 163–177.

[9] Krivulin N. Tropical optimization problems. Advances in economics and optimization: collected scientific studies dedicated to the memory of L. V. Kantorovich, economic issues, problems and perspectives. New York: Nova Science Publishers; 2014. p. 195–214.

[10] Krivulin N. Extremal properties of tropical eigenvalues and solutions to tropical optimization problems. Linear Algebra Appl. 2015;468:211–232.

[11] Krivulin N. A multidimensional tropical optimization problem with a non-linear objective function and linear constraints. Optimization. 2015;64(5):1107–1129.

[12] Krivulin N. Using tropical optimization to solve constrained minimax single facility location problems with rectilinear distance. Comput Manage Sci. 2017;14(4):493–518.

[13] Akian M, Gaubert S, Guterman A. Tropical polyhedra are equivalent to mean payoff games. Int J Algebra Comput. 2012;22(01):1250001.

[14] Gaubert S, Katz RD, Sergeev S. Tropical linear-fractional programming and parametric mean payoff games. J Symb Comput. 2012;47(12):1447–1478.

[15] Krivulin N, Zimmermann K. Direct solutions to tropical optimization problems with nonlinear objective functions and boundary constraints. In: Biolek D, editor. Mathematical methods and optimization techniques in engineering. WSEAS Press; 2013. p. 86–91.

[16] Krivulin N. Direct solution to constrained tropical optimization problems with application to project scheduling. Comput Manage Sci. 2017;14(1):91–113.

[17] Allamigeon X, Gaubert S, Goubault E. The tropical double description method. In: Jean-Marie M, Schwentick T, editors. Proceedings of the 27th International Symposium on Theoretical Aspects of Computer Science – STACS 2010. France: Nancy; 2010. p. 47–58.

[18] Gaubert S, Gunawardena J. The Perron–Frobenius theorem for homogeneous, monotone functions. Trans AMS. 2004;356(12):4931–4950.

[19] Gaubert S, Gunawardena J. The duality theorem for min–max functions. C.R.A.S. Paris, Série I. 1998;326:43–48.

[20] Ehrenfeucht A, Mycielski J. Positional strategies for mean payoff games. Int J Game Theory. 1979;8(2):109–113.

[21] Joswig M, Kulas K. Tropical and ordinary convexity combined. Adv Geom. 2010;10(2):333–352.

[22] De La Puente MJ, Clavería PL. Volume of alcoved polyhedra and Mahler conjecture. Proceedings of the 2018 ACM international symposium on symbolic and algebraic computation; 2018. p. 319–326.

[23] Dhingra V, Gaubert S. How to solve large deterministic games with mean payoff by policy iteration. Proceedings of the 1st International Conference on Performance Evaluation Methodolgies and Tools. VALUETOOLS. Vol. 180; 2006. Article No. 12.

[24] Cuninghame-Green RA, Butkovic P. The equation $A \otimes x = B \otimes y$ over (max, +). Theor Comput Sci. 2003;293(1):3–12.

Appendices

Appendix 1. Proof of Proposition 3.2

We first represent Problem 3.1 equivalently as

$$\min_{x \in \mathbb{R}^n, \lambda \in \mathbb{R}} \{\lambda: \ x^- \otimes p \oplus q^- \otimes x \leq \lambda, \ g \leq x \leq h, \ U \otimes x \leq x\}. \tag{A1}$$

By multiplying the inequality $U \otimes x \leq x$ by U, we deduce that $U^{\otimes 2} \otimes x \leq U \otimes x \leq x$, and hence, by continuing to multiply the inequality by U we determine that $U^* \otimes x \leq x$. Conversely, $U^* \otimes x \leq x$ implies that $U \otimes x \leq x$. Hence, we can restate (A1) as follows

$$\min_{x \in \mathbb{R}^n, \lambda \in \mathbb{R}} \{\lambda: \ p \leq \lambda \otimes x, \ q^- \otimes x \leq \lambda, \quad g \leq x, \ h^- \otimes x \leq 0, \ U^* \otimes x \leq x\}. \tag{A2}$$

Figure A1 provides a mean-payoff game representation of this problem. Here, the group of n nodes of Min (in a circle) corresponds to n variables and the lone-standing node of Min (circle in the bottom) corresponds to the free column. There are two individual nodes of Max corresponding to $q^- \otimes x \leq \lambda$ and $h^- \otimes x \leq 0$. The remaining $3n$ nodes of Max are split into three groups of n nodes: 1) the group on the top ($U^* \otimes x \leq x$), 2) the group on the left ($g \leq x$) and 3) the group on the right ($p \leq \lambda \otimes x$). It is also agreed that an arc between two nodes exists if and only if the corresponding entry of the vector or the matrix marking the corresponding group of arcs on the diagram is finite.

Observe that at every node at which player Max is active, there is no choice to make as there is only one arc leaving that node. Also, let j' be the node (of one of the $[n]$ groups of nodes of Max on the left and on the right) chosen by Min at the bottom node (corresponding to the free column). Assuming this choice of Min and examining the mean-payoff game diagram we see that the total weight of any cycle can be written as $s_{i_1 i_2} + \cdots + s_{i_k i_1}$, where

$$s_{ij} \in \begin{cases} \{-u^*_{ji}\} & \text{if } j \neq j' \\ \{-u^*_{j'i}, q_i - p_{j'} + 2\lambda, q_i + \lambda - g_{j'}, h_i + \lambda - p_{j'}, h_i - g'_j\} & \text{if } j = j', \end{cases} \tag{A3}$$
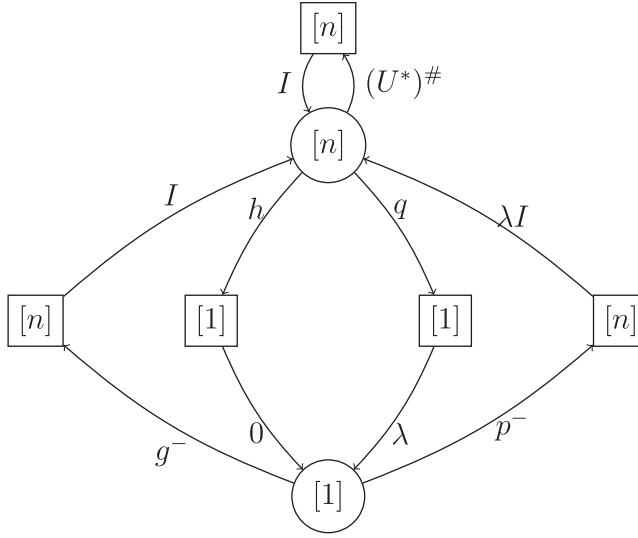
Figure A1. The parametric mean-payoff game corresponding to (A2).

where the above possibilities for s_{ij} are valid only if all matrix and vector entries that take part in them are finite. Hence we obtain that the optimal value of (A2) is equal to the least value of λ such that

$$0 \leq \min_{j'} \min_{i_1,\dots,i_k} \{s_{i_1 i_2} + \cdots s_{i_k i_1}\} \tag{A4}$$

where s_{ij} can take the values described in (A3).

As at every node of player Max there is no choice, we can delete these nodes and aggregate the weights. Also, we can omit the cycles whose weights are composed entirely from the entries of $(U^*)^\sharp$, as these cycle weights do not depend on λ and are nonnegative. We are then left with the cycles that go through the node of Min in the bottom of the diagram and hence also through j'. Note that, as node j' can appear in a cycle only once, there will be at most one occurrence of the second case, where all terms except for the first one come from the arcs in the lower part of the diagram (going to node $n+1$ of Min and back). All other s_{ij} can be compressed to an entry of $(U^*)^\sharp$ by using inequality

$$u^*_{i_1 i_2} + \cdots + u^*_{i_{k-1} i_k} \leq u^*_{i_1 i_k},$$

valid since $(U^*)^{\otimes(k-1)} = U^*$ for each $k > 1$. Thus we have to consider all 3-cycles of the following form:
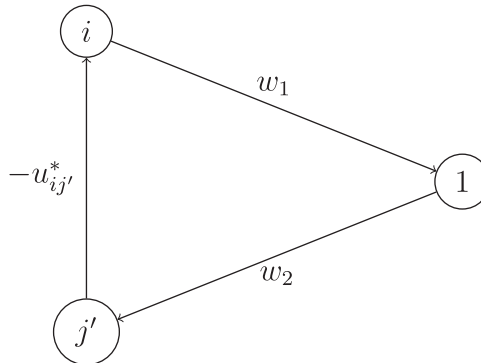
Here $w_1 + w_2$ is one of the finite values in $\{g_i - p_{j'} + 2\lambda,\ q_i + \lambda - g_{j'},\ h_i + \lambda - p_{j'},\ h_i - g_{j'}\}$, and i is such that $u^*_{ij'}$ and one of these values are finite.

So we need to find the minimal λ such that:

(1) $-u^*_{ij'} + q_i - p_{j'} + 2\lambda \geq 0$ for all i and j' such that $u^*_{ij'}, q_i$ and $p_{j'}$ are finite: this is equivalent to $\lambda \geq (q^- \otimes U^* \otimes p)^{\otimes \frac{1}{2}}$;

(2) $-u^*_{ij'} + q_i + \lambda - g_{j'} \geq 0$ for all i and j' such that $u^*_{ij'}, q_i$ and $g_{j'}$ are finite: equivalent to $\lambda \geq q^- \otimes U^* \otimes g$;

(3) $-u^*_{ij'} + h_i + \lambda - p_{j'} \geq 0$ for all i and j' such that $u^*_{ij'}, h_i$ and $p_{j'}$ are finite: equivalent to $\lambda \geq h^- \otimes U^* \otimes p$;

(4) $-u^*_{ij'} + h_i - g_{j'} \geq 0$ for all i and j' such that $u^*_{ij'}, h_i$ and $g_{j'}$ are finite: this is always satisfied by the problem assumption $U^* \otimes g \leq h$.

Hence the optimal value of λ is equal to $(q^- \otimes U^* \otimes p)^{\otimes \frac{1}{2}} \oplus h^- \otimes U^* \otimes p \oplus q^- \otimes U^* \otimes g$, as claimed.

We now deduce the representation of solution set (this part of the proof is similar to that of Krivulin [7, Theorem 6]). The solution set is given by the same inequalities as in (A1) but for the optimal value $\lambda = \theta$, so it is the (finite part of the) alcoved polyhedron described by the following inequalities:

$$x^- \otimes p \leq \theta, \quad q^- \otimes x \leq \theta$$
$$g \leq x, \quad x \leq h, \tag{A5}$$
$$U \otimes x \leq x$$

The first inequality can be rewritten as $p \leq \theta \otimes x$ and the second inequality can be rewritten as $x \leq \theta \otimes q$, and then the first four inequalities of (A5) can be merged into

$$g \oplus \theta^- \otimes p \leq x \leq \theta \otimes q \oplus' h. \tag{A6}$$

It remains to prove that a finite x satisfies (A6) and $U \otimes x \leq x$ if and only if x is as in (5).

Assume first that a finite x satisfies (A6) and $U \otimes x \leq x$. The latter inequality is equivalent to $U^* \otimes x = x$, and therefore from the right-hand side of (A6) we have $U^* \otimes x \leq \theta \otimes q \oplus' h$ and, using Proposition 2.1, $x \leq (U^*)^\sharp \otimes' (\theta \otimes q \oplus' h)$. As we also have $x \geq g \oplus \theta^- \otimes p$ from the left-hand side of (A6), we obtain that $x = U^* \otimes v$, where $v = x$ satisfies

$$g \oplus \theta^- \otimes p \leq x = v \leq (U^*)^\sharp \otimes' (\theta \otimes q \oplus' h),$$

hence x is as in (5).

Now assume that x is as in (5). Since $x = A^* \otimes v$ and $A \otimes A^* \leq A^*$, it satisfies $A \otimes x \leq x$. Since v is finite, $x = U^* \otimes v$ is also finite. We also have $g \oplus \theta^- \otimes p \leq x$ since $U^* \otimes v \geq v$, and $x = U^* \otimes v \leq \theta \otimes q \oplus' h$ follows by Proposition 2.1.

Appendix 2. Proofs of optimality and unboundedness certificates

The proofs written below work both in pseudolinear and in pseudoquadratic case.

A.1. Proof of Proposition 3.5

λ^* is optimal if and only if $\Phi(\lambda^*) = 0$ and $\Phi(\lambda) < 0$ for all $\lambda < \lambda^*$. By the left-hand side of (9), $\Phi(\cdot)$ is a pointwise minimum of a finite number of continuous non-decreasing functions $\Phi_\tau(\cdot)$. Therefore, the above property implies that λ^* is optimal if and only if there exists

a strategy τ of Min such that $\Phi_\tau(\lambda^*) = 0$ and $\Phi_\tau(\lambda) < 0$ for all $\lambda < \lambda^*$. We can further find j and small enough ϵ such that $\Phi_\tau(\lambda) = \chi_j(A_\tau^\sharp B(\lambda)) = a\lambda + b$ for all $\lambda \in [\lambda^* - \epsilon, \lambda]$ and some $a > 0$.

Since $\chi_j(A_\tau^\sharp B(\lambda))$ is the maximal cycle mean (per turn) over all the cycles accessible from the node j of Min in the reduced game defined by A_τ and B, having $\Phi_\tau(\lambda^*) = 0$ means that in the mean-payoff game defined by A_τ and $B(\lambda^*)$ all cycles accessible from the node j of Min have non-positive weight and at least one of them has zero weight. Furthermore, $\chi_j(A_\tau^\sharp B(\lambda)) = a\lambda + b$ with some $a > 0$ and b for all $\lambda \in [\lambda^* - \epsilon, \lambda^*]$ if and only if all cycles of zero weight accessible from the node j of Min contain one of the nodes of player Max that is not from the $[m]$ group on the left of Figure 2 or Figure 8. Thus we have deduced that existence of a strategy τ with the claimed properties is equivalent to the optimality of λ^*.

A.2. Proof of Proposition 3.6

The problem is unbounded if and only if $\Phi(\lambda) \geq 0$ for all $\lambda \in \mathbb{R}$. Since $\Phi(\lambda) = \max_\sigma \Phi^\sigma(\lambda)$, this condition is equivalent to the following:

$$\forall \lambda \in \mathbb{R} \, \exists \sigma \quad \text{s.t.} \, \Phi^\sigma(\lambda) \geq 0, \tag{A7}$$

which can occur if and only if there exist σ and $\lambda' \in \mathbb{R}$ such that $\Phi^\sigma(\lambda) = a$ for all $\lambda \leq \lambda'$ and some constant $a \geq 0$ since function $\Phi^\sigma(\lambda)$ is non-decreasing, piecewise-linear and continuous, which also yields that such σ can make $\Phi(\lambda) \geq 0$ hold for all $\lambda \in \mathbb{R}$. Thus condition (A7) can equivalently be written as:

$$\exists \sigma \quad \text{s.t.} \, \Phi^\sigma(\lambda) \geq 0 \quad \forall \lambda \in \mathbb{R}.$$

Now we use that $\Phi^\sigma(\lambda) = \min_i \chi_i(A^\sharp B^\sigma(\lambda)) = \min_i \min_\tau \Phi_{A,B(\lambda)}(i, \tau, \sigma)$ to rewrite the above condition as follows:

$$\exists \sigma \quad \text{s.t.} \, \forall i, \quad \forall \tau \, \Phi_{A,B(\lambda)}(i, \tau, \sigma) \geq 0 \quad \forall \lambda \in \mathbb{R}, \tag{A8}$$

which is equivalent to saying that the weights of all cycles in the digraph defined by A and $B^\sigma(\lambda)$ are nonnegative and cannot contain λ. Obviously, this condition does not depend on the value of λ, which can be set to 0 as in the claim.

Note that the weight of a cycle must contain λ if and only if such cycle contains a node of Max, which is not from the $[m]$ group (on the left of Figure 2 or 8) since any other node of Max has unique outgoing arc with weight λ. Therefore, condition (A8) holds if and only if any cycle is avoiding these nodes of Max and all cycles have a nonnegative weight.