SF2827 Topics in Optimization:

The linear complementarity problem: Methods and applications

Daniel Olsson, F-06
daolsson@kth.se
860814-0391

Spring 2010

Abstract

This report gives an overview of the linear complementarity problem, as a special case of mathematical programming with equilibrium constraints. The interior point method and two versions of Lemke's method are reviewed, as methods for solving the linear complementarity problem. The report concludes with a description of the Stackelberg game and the bimatrix game as possible applications of mathematical programming with equilibrium constraints and the linear complementarity problem, and also considers Lemke-Howson's algorithm for solving the bimatrix game.

Contents

1 Introduction

The purpose of this report is to look closer at the subclass of mathematical programming with equilibrium constraints (MPEC) that involves the linear complementarity problem (LCP) and consider different ways of solving them. Before giving a description of the LCP, a brief presentation of the MPEC and the variational inequality problem will be given. The equilibrium constraints of the MPEC are normally expressed as parametric variational inequality problems. As will be shown, the linear complementarity problem is a special case of the variational inequality problem, and thus also a special case of the MPEC.

The detailed outline of the report is as follows. First, a description of the MPEC will be given in section 2. The problem formulation and description of the linear complementarity problem will be given in section 3. Two methods for finding solutions to LCPs will be considered in section 4: An interior point method developed by Kojima, and Lemke's method. Finally, Stackelberg games and bimatrix games will be defined and related to MPEC and LCP in section 5. Also, the Lemke-Howson method for finding equilibrium points in the bimatrix game will be presented.

2 Mathematical Programming with Equilibrium Constraints

2.1 Overview

A mathematical program with equilibrium constraints (MPEC) is defined as "an optimization problem in which the essential constraints are defined by a parametric variational inequality or complementarity system" [10]. MPEC is a quite new field of research and is an extension of bilevel programming (BP). The equilibrium constraints are normally expressed either as a complementarity system or as a variational inequality, where the former is a special case of the latter. MPEC has a wide range of applications, for example in economics and on the electricity market. The concept of MPEC has its origin in the economic concept of Stackelberg games. Stackelberg games and bimatrix games will be discussed more in detail in section 5.

MPEC is a hierarchical program in two levels. It is sometimes referred to as Mathematical Programming with Complementarity Constraints (MPCC), due to the complementarity form of the equilibrium constraints. The complementarity constraints can further be divided into Mixed Complementarity Programs (MCP) and Generalized Linear Complementarity Programs (GLCP). The GLCP, also known as the linear complementarity problem over cones, unifies the problem classes of the monotone linear complementarity problem, the linear program, convex quadratic programs and the mixed monotone LCP [5].

2.2 Problem formulation

The mathematical formulation of the MPEC is as follows. Let f and F be the mappings $f : \mathbb{R}^{n+m} \to \mathbb{R}$ and $F : \mathbb{R}^{n+m} \to \mathbb{R}^m$, let $Z \subseteq \mathbb{R}^{n+m}$ be a nonempty closed set, and let $C : \mathbb{R}^n \to \mathbb{R}^m$ be a set-valued map with (possibly empty) closed convex values. The set of all vectors $x \in \mathbb{R}^n$ for which $C(x) \neq \emptyset$ is the domain of C and denoted $\text{dom}(C)$. Let X be the projection of Z onto $\mathbb{R}^n$, so that

$$X = \{x \in \mathbb{R}^n : (x, y) \in Z, \text{ for some } y \in \mathbb{R}^m\}.$$

The set $C(x)$ defines the values of y that are feasible for each given $x \in X$. Z is the feasible region for the pair (x, y) of the upper-level problem. We assume that $X \subseteq \text{dom}(C)$. The MPEC typically involves two variables, $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$, where some of the constraints are parametric variational inequalities or a complementarity system, in which y are the so called primary variables and x is the parameter vector. With the above definitions, the general formulation of the MPEC optimization problem can be expressed as

$$\begin{aligned} \underset{x,y}{\text{minimize}} \quad & f(x, y) \\ \text{subject to} \quad & (x, y) \in Z \\ & y \in \mathcal{S}(x) \end{aligned} \tag{2.1}$$

where for each $x \in X$, $\mathcal{S}(x)$ is the set of solutions for a variational inequality. As will be seen in section 3.1, the linear complementarity problem is a special case of the variational inequality problem. The function f is the objective function of the upper-level problem. F is the equilibrium function of the inner-problem. Computationally this general form of the problem is very hard to solve. The mathematical program with equilibrium constraints are generally NP-hard. There are alternative ways of expressing the MPEC problem. A more complementarity-oriented formulation is,

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to} \quad & c(x) \geq 0 \\ & 0 \leq x_1 \perp x_2 \geq 0 \end{aligned} \tag{2.2}$$

where $\perp$ is the complementarity operator, i.e for each component pair i of the vectors x_1 and x_2, if $x_{1i} \neq 0$, then $x_{2i} = 0$. MPEC problems can be transformed into a nonlinear program (NLP) by replacing the complementarity constraint $0 \leq x_1 \perp x_2 \geq 0$ with the constraint $X_1 x_2 \leq 0$, where

$X_1 = diag(x_1)$. This gives the NLP formulation of the problem:

$$\begin{array}{ll} \underset{x}{\text{minimize}} & f(x) \\ \text{subject to} & c(x) \geq 0 \\ & x_1, x_2 \geq 0, \\ & X_1 x_2 \leq 0 \end{array} \tag{2.3}$$

There are serious difficulties with solving (2.3), mainly because standard stability assumptions are not met. However, there has been some success in finding local optimal points for (2.3) using sequential quadratic programming methods [9].

3 The Linear Complementarity Problem (LCP)

3.1 Problem formulation

The Linear Complementarity Problem (LCP) is defined in the following way.

Definition 3.1 (The Linear complementarity problem)**.** *Let w be a mapping $w : \mathbb{R}^n \to \mathbb{R}^n$. Given w, one seeks a vector $z \in \mathbb{R}^n$ such that*

$$w = Mz + q, \quad z \geq 0, \; w \geq 0, \; z_i w_i = 0 \tag{3.1}$$

for $i = 1, 2, \ldots, n$.

Using shorter notation, the linear complementarity problem defined above can be expressed as the LCP (q, M). From the constraints $z \geq 0$, $w \geq 0$ and $z_i w_i = 0$ follows that z and w are required to be nonnegative, and that at least one of the component-pair z_i, w_i must be zero, e.g. if $z_1 \neq 0$, then $w_1 = 0$. This complementarity constraint can also be expressed as $0 \leq z \perp w \geq 0$, where $\perp$ is the complementarity operator. As was mentioned earlier, the equilibrium constraints of the MPEC are expressed by variational inequalities. The variational inequality (VI) contains a broader class of problems than the LCP. The standard mathematical formulation of the variational inequality problem is the following.

Definition 3.2 (The Variational inequality problem)**.** *Determine a vector $z \in K \subset \mathbb{R}^n$, such that*

$$(y - z)^T F(z) \geq 0, \quad \text{for all } y \in K \tag{3.2}$$

where F is a given continous function from $K \to \mathbb{R}^n$ and K is a given closed convex set.

Using shorter notation, the variational inequality problem can be stated as VI (K, F). The LCP is a special case of the variational inequality [4, "Linear Complementarity Problem"], which can be seen by setting $y = 0$ and demanding that z and $F(z)$ are nonnegative, i.e. in $\mathbb{R}_+$. The resulting constraints $F(z) \geq 0$, $z \geq 0$ and $z^T F(z) \leq 0$ can be expressed with the complementarity operator as

$$0 \leq z \perp F(z) \geq 0.$$

which is the complementarity constraint of the LCP. From the LCP being a special case of the variational inequality problem, it can be shown that the LCP (q, M) and the VI $(Mz + q, \mathbb{R}_+)$ have the same solutions.

Theorem 3.1. *Let $F = Mz + q$, $M \in \mathbb{R}^{n\times n}$, $q \in \mathbb{R}^n$, $z \in \mathbb{R}^n_+$. Then the VI($F, \mathbb{R}_+$) and the linear complementarity problem LCP(q, M) have precisely the same solutions, if any.*

Proof. Suppose that z solves VI$(F, \mathbb{R}_+)$. By taking $y = 0$ in (3.2), we obtain

$$z^T F(z) \leq 0. \tag{3.3}$$

Since $z \in \mathbb{R}_+$, it follows that also $2z \in \mathbb{R}_+$. Thus by inserting $y = 2z$ in (3.2), we obtain

$$z^T F(z) \geq 0. \tag{3.4}$$

Combining the inequalities (3.3) and (3.4), we get

$$z^T F(z) = 0. \tag{3.5}$$

which in turn yields

$$y^T F(z) \geq 0$$

for all $y \in \mathbb{R}_+$. In the same way, if z solves the LCP(q, M), then it can be shown that z solves the VI$(F, \mathbb{R}_+)$. □

It can be noted that theorem 3.1 applies to complementarity problems in general, i.e. also the nonlinear complementarity problem has the same solutions as the variational inequality problem. However, in this report we are mainly concerned with the LCP. A generalized proposition for the complementarity problem can be found in [3, pp. 4-5].

The description of the variational inequality problem shows the relation between the MPEC and the LCP. As was mentioned, the equilibrium constraints of the MPEC are expressed as variational inequalities. Thus, from the LCP being a special case of the variational inequality, it follows that a subgroup of MPEC are those problems whose equilibrium constraints are defined by a LCP. In this sense, the LCP is a special case of the MPEC.

3.2 Existence and uniqueness of solutions

The notion of monotonicity is central in the existence and uniqueness of solutions to the linear complementarity problem and the variational inequality problem. As will be discussed in section 4, there are interior point methods that can solve monotone LCPs in polynomal time. Monotone LCPs are therefore considered to be "easy" problems. Some fundamental existence and uniqueness theorems will be stated here. However, before stating the conditions for existence and uniqueness of solutions for the LCP, some definitions are needed.

Definition 3.3 (Principal minor)**.** *Let A be an $n \times n$ matrix. A $k \times k$ submatrix of A formed by deleting $n - k$ rows of A, and the same n - k columns of A, is called a principal submatrix of A. The determinant of a principal submatrix of A is called a principal minor of A.*

Different characteristics of the M-matrix are important when it comes to deciding whether a LCP (q, M) has any solutions. One common matrix class is the class of P-matrices.

Definition 3.4 (P-matrix)**.** *A real square matrix M is a P-matrix if it satisfies*

$$z_i(Mz)_i > 0,$$

for all $0 \neq z \in \mathbb{R}^n$.

It can be shown that a LCP (q, M) has at least one solution for every real vector q if the matrix M is a P-matrix and all its principal minors are positive.

Theorem 3.2. *Let $M \in \mathbb{R}^{m\times m}$ be a P-matrix with all of its principal minors positive. Then the LCP (q, M) has a solution for all vectors $q \in \mathbb{R}^m$.*

As was mentioned earlier, several of the central existence and uniqueness results are closely related to the monotonicity of the problem. There are different types of monotonicity and they play a important role to the existence of solutions for LCPs and VIs. In this report, monotonicity is defined in the following way.

Definition 3.5 (Monotonicity)**.** *A mapping $F : K \subseteq \mathbb{R}^m \to \mathbb{R}^m$ is said to be*

1. *monotone on K if for all pairs $(u, v) \in K \times K$,*

$$(u - v)^T (F(u) - F(v)) \geq 0;$$

2. *strictly monotone on K if for all pairs $(u, v) \in K \times K$ with $u \neq v$,*

$$(u-v)^T(F(u)-F(v)) > 0;$$

3. *strongly monotone on K if there exists a constant $c > 0$, such that for all pairs $(u, v) \in K \times K$,*

$$(u-v)^T(F(u)-F(v)) \geq c \parallel u-v \parallel^2 .$$

The results below are originally from [10, p. 54], in which they are presented for the variational inequality problem. Due to theorem 3.1, the results in theorem 3.3 can be transfered to the LCP case as well.

Theorem 3.3. *Let K be a closed convex set in $\mathbb{R}^m$ and $F : K \rightarrow \mathbb{R}^m$ be a continuous mapping. Let SOL(F, K) denote the (possibly empty) solution set of the VI (F, K).*

1. *If F is monotone on K, then SOL(F, K), if nonempty, is a closed convex set.*

2. *If F is strictly monotone on K, then SOL(F, K) consists of at most one element.*

3. *If F is strongly monotone on K, then SOL(F, K) consists of exactly one element.*

The existence of solution is guaranteed when q is nonnegative. This will be used in Lemke's method described in section 4.3.

Theorem 3.4. *If q is nonnegative, then the LCP (q, M) in (3.1) is always solvable, where $z = 0$ is a trivial solution.*

Proof. The proof of theorem 3.4 is trivial. Simply set $z = 0$ and verify that the constraints in the definition of the LCP holds for all $q \geq 0$. $\square$

4 Methods for the LCP

4.1 Introduction

Quite many algorithms have been developed for solving the linear complementarity problem. The perhaps most well known method for solving LCPs is Lemke's method. One drawback with Lemke's method is its worst-case exponential running time, which makes it inefficient for larger problems. However, there are methods for solving the LCP that run in polynomial time.

The two methods that will be considered in this report are Lemke's method and a polynomial-time interior point method. The interior method presented in this report was developed by Kojima [7]. It has been inspired by the algorithms that have been used for solving linear programs in polynomial-time, originally developed by Karmarkar [6]. For the LCP to be solved in polynomial time with the interior point method requires the matrix M to be positive semi-definite.

4.2 Interior-point method

The interior point method presented in this report has a total computational complexity of $\mathrm{O}(n^{3.5})$. However, modifications to the algorithm have been suggested in [7], which would reduce the total computation complexity to $O(n^3)$ operations.

The following assumptions are made for the LCP (q, M), with $M \in \mathbb{R}^{n\times n}$ and $q \in \mathbb{R}^n$, that is to be solved with the interior point method considered in the report:

- $n \geq 2$. The case $n = 1$ has a trivial solution and will not be considered.
- The elements in M and q are integers.
- M is positive semi-definite, i.e. $z^T Mz \geq 0$ for all $z \in \mathbb{R}$.
- Each row of M has at least one nonzero element.

Before presenting the algorithm, a system of equations that will be used frequently in the algortihm will be defined. Let H be a mapping $H : \mathbb{R}^n_+ \times \mathbb{R}^{2n}_+ \rightarrow \mathbb{R}^n \times \mathbb{R}^n$ defined by

$$H(\mu, z, w) = (ZWe - \mu e, w - Mz - q) \tag{4.1}$$

for all $(\mu, z, w) \in \mathbb{R}^{1+2n}_+$, where $W = \mathrm{diag}(w)$, i.e. the diagonal matrix of w, and e is a n-dimensional vector of ones. It can be shown that (z, w) is a solution to the LCP if and only if it is a solution to equation (4.1) for $\mu = 0$. From this follows that LCP (q, M) is equivalent to

$$H(0, z, w) = 0 \quad \text{and} \quad (z, w) \in \mathbb{R}^{2n}_{+} \tag{4.2}$$

The symbol S is used for the set of all the feasible solutions of the LCP, S_{int} is the interior of S and S_{cp} are all the solutions of the LCP.

$$\begin{aligned} S \;\; &= \{(z, w) \in \mathbb{R}^{2n}_{+} : w = Mz + q\} \\ S_{int} &= \{(z, w) \in \mathbb{R}^{2n}_{++} : w = Mz + q\} \\ S_{cp} &= \{(z, w) \in S : z_i w_i = 0, \quad (i = 1, 2, \ldots, n)\} \end{aligned}$$

The $\mathbb{R}^n_+ = \{z \in \mathbb{R}^n : z \geq 0\}$ is the nonnegative orthant of $\mathbb{R}^n$, and $\mathbb{R}^n_{++} = \{z \in \mathbb{R}^n : z > 0\}$ is the positive orthant of $\mathbb{R}^n$.

Each point $(x, y) \in \mathbb{R}^{2n}$ that satisfies the system (4.1) for some $\mu > 0$ is called a center of the feasible region S. The set of all center points is denoted S_{cen} and is defined as

$$S_{cen} = \{(x, y) \in \mathbb{R}^{2n}_{+} : H(\mu, z, w) = 0 \; for \; some \; \mu > 0\}.$$

The algorithm uses Newton iterations on the system of equations (4.2) for a parameter $\mu \geq 0$ that is made smaller for each iteration until the solution is sufficiently accurate. The accuracy of the solution is controlled by the constant L, which is used in the stopping criteria of the algorithm. In [7], the size of L is defined as

$$L = \lfloor \sum_{i=1}^{n} \sum_{j=1}^{n+1} log(|a_{ij}| + 1) + log(n^2) \rfloor + 1, \tag{4.3}$$

where a_{ij} is the (i, j)th element of the $n \times (n + 1)$ matrix $A = [M \; q]$. Here $\lfloor \xi \rfloor$ is defined as the largest integer not greater than $\xi \in \mathbb{R}_+$.

The values of z and w in the k:th iteration of the algorithm are denoted z^k and w^k, respectively. Assuming that the initial point (z^1, w^1) is known, the steps of the algorithm are:

- Step 0: Let α be a positive constant, $\alpha \leq 0.1$ and $\delta = \alpha/(1 - \alpha)$. Let $k = 1$.
- Step 1: If $(z^k)^T w^k \leq 2^{-2L}$, then stop. Otherwise, go to step 2.
- Step 2: Let $\mu = (1 - \delta/n^{1/2})(z^k)^T w^k/n$ and $(z, w) = (z^k, w^k)$. Define the diagonal matrices Z and W.

- Step 3: Compute the Newton direction $(\Delta z, \Delta w)$ using the following two expressions

$$\Delta z = (\mu + Z^{-1}W)^{-1}(\mu Z^{-1}e - We) \tag{4.4}$$
$$\Delta w = M\Delta z \tag{4.5}$$

  and then compute the new point

$$(z^{k+1}, w^{k+1}) = (z^k, w^k) + (\Delta z, \Delta w). \tag{4.6}$$

- Step 4: Let $k = k + 1$. Go to step 1.

The algorithm is iterated until the stopping criteria has been met. By adjusting L, the obtained solution can be made as accurate as needed. For a description of how to prepare an initial point for the interior point algorithm, see [7].

4.3 Lemke's method: Augmented problem

In this report, Lemke's method will be presented in two versions. The first algorithm solves the augmented LCP $(\overline{q}, \overline{M})$ and is based on the algorithm described in [1, p.267-282]. As will be seen, by fulfilling certain conditions the solution will also be a solution to the original LCP(q, M).

The concepts of "basic" and "nonbasic" variables are central in Lemke's method, much in the same way as they are in the simplex method in linear programming. A basic variables is equivalent to a variable with a linearly independent column and a nonbasic variable is a variable with a linearly dependent column. Thus, the basic variables can be varied, and this variation is observed in the nonbasic variables. The (nonbasic) variable that is being varied is referred to as the "driving variable".

"Pivoting" is another term that is frequently used in Lemke's method. Pivoting is the changing of a basic and a nonbasic variable in such a way that the nonbasic variable becomes basic and the basic variable becomes nonbasic. When a basic and a nonbasic variable are being pivoted, the system of equation is modified in such a way that the new set of nonbasic variables are expressed by the new set of basic variables.

The "blocking variable" is the variable whose constraint is first to be violated as the value of the driving variable is increased.

The augmented LCP is defined as the problem

$$w_0 = q_0 + 0 \cdot z_0 - d^T z \geq 0, \quad z_0 \geq 0, \quad z_0 w_0 = 0, \tag{4.7}$$

$$w = q + d \cdot z_0 + Mz \geq 0, \quad z \geq 0, \quad z^T w = 0. \tag{4.8}$$

In matrix form, the left inequality in (4.7) and (4.8) can be expressed as

$$\begin{bmatrix} w_0 \\ w \end{bmatrix} = \begin{bmatrix} q_0 \\ q \end{bmatrix} + \begin{bmatrix} 0 & -d^T \\ d & M \end{bmatrix} \begin{bmatrix} z_0 \\ z \end{bmatrix}, \tag{4.9}$$

and the augmented M and q can thus be expressed as

$$\overline{M} = \left(\begin{smallmatrix} 0 & -d^T \\ d & M \end{smallmatrix}\right), \quad \overline{q} = \begin{pmatrix} q_0 \\ q \end{pmatrix},$$

where the constant q_0 is nonnegative and sufficiently large and $d \geq 0$. Let $\overline{z}_0 \geq 0$ be the smallest non-negative scalar such that $w = q + dz_0 \geq 0$ for all $z_0 \geq \overline{z}_0$.

$$\overline{z}_0 = \max\{-q_i/d_i\}$$

Note that by making the assumption that q_i is negative for at least one $i \in \{1, \ldots, m\}$, it will follow that $\overline{z}_0 \geq 0$.

Lemma 4.1. *A solution $(\overline{q}, \overline{M})$ for the augmented LCP, where $\overline{z}_0 = 0$, gives a solution to the original LCP (q, M).*

When using Lemke's algorithm for the augmented LCP in computing, the constraints in (4.7) can be replaced by the alternative formulation:

$$w = q + dz_0 + Mz, \quad z_0 \geq 0, \quad z \geq 0, \quad z^T w = 0.$$

The advantage of this formulation is that the requirement on q_0 to be sufficiently large is removed, which makes it more practical in computations. This version of the algorithm is commonly called the streamlined version of Lemke's method and will be described in section 4.4. The notion of nondegeneracy is important to the existence and uniqueness of solution for Lemke's method.

Definition 4.1. *Let $H_{q,M}(z) = min(z, q + Mz)$. A vector $z \in \mathbb{R}^n$ is said to be* nondegenerate *with respect to $H_{q,M}$ if $z_i \neq (q+Mz)_i$ for each $i \in 1, \ldots, n$.*

This version of Lemke's algorithm is used for solving the augmented problem for $\overline{z}_0 = 0$, which will yield a solution for the original LCP as well. Lemke's method for the augmented LCP uses the following procedure:

1. Initialization. The input is the augmented LCP $(\overline{q}, \overline{M})$. If $q \geq 0$, then stop; The LCP (q, M) is solved for $z = 0$. Otherwise, find $\overline{z}_0$, the smallest value for $w = q+dz_0 \geq 0$. Let w_r denote the w-component that equals zero when $z_0 = \overline{z}_0$. Pivot $\langle w_r, z_0\rangle$. This causes the complements w_0 and z_0 to be basic variables, while w_r and z_r becomes nonbasic. Choose z_r, the complement of w_r, to be the new driving variable.

2. Determine the blocking variable. If z_0 is the blocking variable, then pivot $\langle z_0, \text{driving variable}\rangle$. Use the minimum ratio test to determine the blocking basic variable. If w_0 is blocking, then stop.

3. Pivoting. The driving variable is blocked. If z_0 is the blocking variable, then pivot $\langle z_0, \text{driving variable}\rangle$ and stop. A solution to (q, M) has been found. If som other variable blocks the driving variable, then return to step 1, using the complement of the most recent blocking variable as the driving variable.

It will be shown that the algorithm presented above terminates in a finite number of steps, if the augmented problem is nondegenerate.

Definition 4.2. *A vector (z_0, z) is said to be* almost complementary *with respect to the equation (4.9) if the vector satisfies the following conditions:*

$$z_0 w_0 > 0 \text{ and } z_i w_i = 0 \quad \text{for all } i \neq 0$$

Lemma 4.2. *When w_0 and z_0 are basic in Lemke's algorithm, the column of the driving variable contains at least one negative entry.*

A proof of lemma 4.2 can be found in [1, p. 268].

Theorem 4.1 ([1] p. 270)**.** *When applied to a nondegenerate augmented problem $(\overline{q}, \overline{M})$, Lemke's algorithm terminates in finitely many steps.*

Proof. By lemma 4.2, a suitable pivot entry is available at every iteration. The nondegeneracy assumption implies that the algorithm generates a unique almost complementary path. The almost complementary extreme points of the feasible region for $(\overline{q}, \overline{M})$ that occur along this path correspond to almost complementary nonnegative basic solutions of (4.7) and (4.8).

There are at most two almost complementary edges of the path incident to an almost complementary extreme point of the feasible region to $(\overline{q}, \overline{M})$. These edges can be taken away by making one of the members of the nonbasic pair increase from the value 0. The nondegeneracy assumption guarantees that all nonbasic variables can be made positive before a basic variable decreases to zero.

The almost complementary path cannot return to a previously encountered almost complementary extreme point. If it would, there would have to be at least three almost complementary edges incident to it. For a given problem, there can only exist a finite number of any kind of bases, and in particular, there must be a finite number of almost complementary bases. Therefore, the algorithm must terminate in a finite number of steps.

□

4.4 Lemke's method: Streamlined version

There is a second version of Lemke's algorithm that is more practical when it comes to computing, by some referred to as the "streamlined" version of Lemke's method. In comparison Lemke's method for the augmented problem, the main difference is that the constraint

$$w_0 = q_0 - d^T z \geq 0$$

is removed, and thus the requirement to choose a sufficiently large constant q_0 disappears. The definition of this problem is

$$w = q + dz_0 + Mz \geq 0, \quad z_0 \geq 0, \quad z \geq 0, \quad z^T w = 0. \tag{4.10}$$

This system will be denoted as (q, d, M). The solution to the original LCP can be obtained through the solution (q, d, M) when $z_0 = 0$. Without the constraint $w_0 \geq 0$, it is possible that the column of a driving variable could be nonnegative, which might cause the variable to be unblocked. If no blocking variable is found by increasing the driving variable, a second ray is generated. A secondary ray is an almost complementary edge of the feasible region of (q, d, M) that is necessarily unbounded. The algorithm has the following form:

1. Initialization, Input (q, d, M). If $q \geq 0$, then stop and return $z = 0$ as the solution to (q, M). Otherwise, let $\overline{z}_0$ be the smallest value of the artificial variable z_0 for which $w = q + dz_0 \geq 0$. Let w_r denote w that equals zero when $z_0 = \overline{z}_0$. Pivot $\langle w_r, z_0 \rangle$. After this pivot, the complementary variables w_r and z_r are both nonbasic. Let the driving variable be z_r, i.e. the complement of w_r.

2. Determination of the blocking variables (if any). If the column of the driving variable has at least one negative entry, use the minimum ratio test to determine the basic variable that blocks the increase of the driving variable. If the driving variable is unblocked, then stop.

3. Pivoting. The driving variable is blocked. If the driving variable is being blocked by z_0, then pivot

   $$\langle z_0, \textit{driving variable} \rangle$$

   and stop. A solution to (q, M) has been found. If some other variable blocks the driving variable, then pivot

   $$\langle \textit{blocking variable}, \textit{driving variable} \rangle.$$

   Return to Step 2, using the complement of the most recent blocking variable as the new driving variable.

As in the case with Lemke's method for the augmented problem, the streamlined version of the method will terminate in finitely many steps.

Lemma 4.3. *When applied to a nondegenerate LCP (q, d, M), the algorithm will terminate in finitely many steps with either a secondary ray or else a complementary feasible solution of (q, d, M) and thus a solution to (q, M).*

Lemma 4.4. *If the algorithm terminates with a secondary ray when applied to (q, d, M), then M reverses the sign of some nonzero nonnegative $\tilde{z}$, so that*

$$\tilde{z}_i(M\tilde{z})_i \leq 0, \tag{4.11}$$

for $i = 1, \ldots, n$.

Proofs of the lemmas 4.3 and 4.4 can be found in [1, pp. 274-277]. An assumption that is made is that the vector d is strictly positive.

Theorem 4.2. *The algorithm will solve any nondegenerate LCP (q, M) if M is a P-matrix.*

Proof. Lemma 4.3 stated that the algorithm will terminate either with a secondary ray or else as a solution to (q, M), if (q, d, M) is nondegenerate. If the algorithm terminates with a secondary ray, then according to lemma 4.4, M reverses the sign of some nonnegative element $\tilde{z}$. However, if the matrix M is a P-matrix, then the definition of the P-matrix gives that sign cannot be reversed. Therefore, if M is a P-matrix, then the algorithm cannot terminate with a secondary ray. Thus, for a nondegenerate LCP in which M is a P-matrix, the algorithm will terminate with a feasible solution to (q, d, M) and thus with a solution to the original LCP (q, M). □

It can further be noted that the algorithm can solve degenerate LCPs as well, under certain circumstances. The risk of degenerate LCPs is that the algorithm will cycle instead of terminate in finitely many steps. Different methods have been developed to deal with cycling. Two possible approaches for Lemke's method are lexicographic degeneracy resolution and least-index degeneracy resolution. Descriptions of these so called anti-cycling methods can be found in [1, pp. 336-352]

5 Applications

5.1 Introduction

In this section bimatrix games and Stackelberg games will be reviewed. The Stackelberg game is closely related to the MPEC and is an extension of the concept of the Nash game, in which a number of players each have a set of strategies to choose from.

In a Nash game, cooperation is not allowed, and the players choose strategies independently from each other. Each player is assumed to choose the strategy which is expected to produce the greatest profit for himself. In a Stackelberg game, there is one player who acts as a leader while the other players are followers. A brief introduction to Nash and Stackelberg games can be found in [11].

Assume there are N players and that the payoff function for the i:th player is $g_i(x)$, for $i = 1, \dots, N$, where $x = (x_1, \dots, x_N)$ contains the decision variables and x_i is the decision variable for the i:th players. The strategy space for the i:th player is given by $X \subset \mathbb{R}^{n_i}$, where n_i is an integer and signifies the number of strategies that player i has to choose between. Let $g_i : \mathbb{R}^{n_i+n_N} \to \mathbb{R}$ and the player i's decision variable $x_i \in X_i$ for $i = 1, \dots, N$. In the Nash game, player i's objective function $g_i(x)$ is expressed as

$$\underset{x_i \in X_i}{\text{maximize}}\ g_i(x) \qquad i = 1, \dots, N. \tag{5.1}$$

In other words, each player tries to maximize its own objective function. However, the value of the objective function of a player depends on the action of all player, not just one.

In a Stackelberg game, player 1 is a "leader" and therefore gets to act first. The leader's choice of strategy is affected by how the other players are expected to react to the leader's action. The other players will try to find an action that maximizes their profit under the circumstances that are imposed by the leader's choice of action. The leader is denoted by index 1 and the other players are here denoted by the index -1, i.e. $x_{-1} = (x_2, \dots, x_N)$.

The Stackelberg game is mathematically similar to the Nash game, but with a new objective function and additional constraints:

$$\underset{x_1 \in X_1}{\text{maximize}}\ g_1(x) \tag{5.2}$$

$$\text{subject to } x_{-1} \in \text{argmax}\{g_{-1} : x_{-1} \in X_{-1}\}. \tag{5.3}$$

In the Stackelberg game, the leader seeks to maximize its objective function. This objective function is subject to the constraint that, in response to the leader's action, the followers will choose an action that maximizes their own individual objective functions.

The Stackelberg game is a bilevel optimization problem in the sense that there is an upper-level problem (the leader, acting first, tries to maximize its profit) and a lower-level problem (the followers, acting after the leader, seek to maximize their profit). Typically, the leader can be a large dominating firm in some market, while the followers are smaller, competing firms.

The problem formulation in (5.2) and (5.3) is an MPEC, where the expression (5.3) has the same role as the constraint $y \in \mathcal{S}(x)$ in (2.1). In contrast, a bimatrix game is a two player non-zero sum game, meaning that the gain of one player does not have to be equal to the loss of the other player. Bimatrix games will be looked closer at in section 5.2.

5.2 Bimatrix games

An early example of application of the linear complementarity problem is bimatrix games. Bimatrix games are found in game theory and consist of two players. The aim of each player is to find the strategy that returns the highest profit (or the lowest cost) for that player. Both participants control some, but not all, of the possible actions. The strategy of each player is either a pure strategy or a randomized strategy. A pure strategy means that the player will choose the same action every time the game is played. A randomized strategy means that each possible action has a probability assigned to it, the probability that the player will choose that action.

A further description of the bimatrix game can be found in [2, pp.277-298] and [8].

A bimatrix game can be defined in the following way. Let $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{m\times n}$. The cost for player one is given by the element a_{ij} in matrix A and the cost for player two is given by b_{ij} in the matrix B, where i is player one's pure strategy and j is player two's pure strategy, $i = 1, \dots, m$, $j = 1, \dots, n$. The element x_i in player one's randomized strategy vector $x \in \mathbb{R}^m$ is the probability that player one will choose the pure strategy i. Player two's randomized strategy vector is $y \in \mathbb{R}^n$. From this naturally follows that

$$x \geq 0, \quad \sum_{i=1}^{m} x_i = 1,$$
$$y \geq 0, \quad \sum_{j=1}^{n} y_j = 1.$$

Thus, the expected cost will be $x^T A y$ for player one and $x^T B y$ for player two. The bimatrix game can be expressed as $\Gamma(A, B)$, where A and B are the cost matrices of player one and player two, respectively.

As an example, consider the bimatrix game $\Gamma(A, B)$, where

$$A = B^T = \begin{bmatrix} 10 & 20 \\ 30 & 15 \end{bmatrix}. \tag{5.4}$$

The element a_{ij}, on row i and column j in matrix A, determines the cost for player one. The cost for player two is determined by the element b_{ij}, located on the same position in matrix B. Here i represents the i:th element of player ones strategy set, and j represents the j:th element in the strategy set of player two. In other words, i and j are the strategies that player one and player two choose, respectively. The example problem (5.4) was originally given in [1, p. 287]. The solutions to problem will be given in section 5.3.

A situation in which neither player gains anything from changing its randomized strategy, assuming that the opponent does not change strategy as well, is called a Nash equilibrium.

Definition 5.1 (Nash Equilibrium)**.** *A randomized strategy pair (x^*, y^*), where $x^* \in \mathbb{R}^m$, $y^* \in \mathbb{R}^n$, is called a Nash Equilibrium if*

$$(x^*)^T A y^* \leq x^T A y^* \textit{ for all } x \geq 0 \quad \textit{and} \quad \sum_{i=1}^{m} x_i = 1, \tag{5.5}$$

$$(x^*)^T B y^* \leq x^T A y \textit{ for all } y \geq 0 \quad \textit{and} \quad \sum_{i=1}^{n} y_i = 1. \tag{5.6}$$

The bimatrix game $\Gamma(A, B)$ can be transformed into a LCP by making the assumption that the elements of A and B are positive. The equilibrium solutions of the bimatrix game are not affected by this assumption; if some elements would be negative in A or B, a sufficiently large scalar can be added to all elements in both matrices so that the matrices become positive. The LCP would have the form

$$u = -e_m + Ay \geq 0, \quad x \geq 0, \quad x^T u = 0, \tag{5.7}$$

$$v = -e_n + B^T x \geq 0, \quad y \geq 0, \quad y^T v = 0, \tag{5.8}$$

where e_m and e_n are vectors containing only ones, of length m and n, respectively. If (x^*, y^*) is a Nash equilibrium, then (x', y') is a solution to (5.7), where

$$x^* = x'/e_m^T x',$$
$$y^* = y'/e_n^T y',$$

where x' and y' need to be non-zero. The assumption that A and B are positive matrices implies that x' and y' are nonnegative. The q-vector and M-matrix for the LCP(q, M) would be written as

$$q = \begin{bmatrix} -e_m \\ -e_n \end{bmatrix}, \quad M = \begin{bmatrix} 0 & A \\ B^T & 0 \end{bmatrix}. \tag{5.9}$$

The resulting problem can then be written in the LCP form

$$w = Mz + q, \quad z \geq 0, \quad w \geq 0, \quad z^T w = 0,$$

with

$$z = \begin{bmatrix} x \\ y \end{bmatrix} \text{ and } w = \begin{bmatrix} u \\ v \end{bmatrix}.$$

5.3 Solving the bimatrix game

The Lemke-Howson algorithm is a method used for solving bimatrix games. The algorithm, which is also described in [1, p. 284-287], is combinatorial and finds a Nash equilibrium in the bimatrix game. As can be seen, it is similar to the previously described Lemke's method used for solving LCPs.

1. Initialization. Use the LCP (q, M) of order $m+n$ as input, with q and M as in (5.9). Select an index $k \in \{1, \ldots, m\}$. Let $s \in arg\ min_{1 \leq j \leq n}\ b_{kj}$. Pivot $\langle x_k, v_s \rangle$. This yields an almost complementary, but infeasible solution. Let $r \in arg\ min_{1 \leq i \leq m} a_{is}$. Pivot $\langle u_r, y_s \rangle$. The solution is now almost complementary and feasible. The basic pair is (x_k, u_k), and the nonbasic pair is (u_r, x_r). If $r = k$, then stop: A solution has been found. Otherwise, let x_r be the driving variable.

2. Determine the blocking variable (if any). Use the minimum ratio test to determine whether there is a basic variable that blocks the increase of the driving variable. If not, stop.

3. Pivoting. The driving vaiable is blocked. Pivot

$$\langle blocking\ variable, driving\ variable \rangle$$

   If the blocking variable belongs to the basic pair, a solution to (q, M) is at hand. Otherwise return to Step 2 using the complement of the most recent blocking variable as the new driving variable.

Theorem 5.1. *The Lemke-Howson algorithm finds a solution of every non-degenerate LCP corresponding to a bimatrix game.*

Proof. The proof is by contradiction. If a solution is not found in Step 1, then an almost complementary extreme point of the feasible set given by

$$\begin{aligned} u &= -e_m + Ay \geq 0, \quad x \geq 0, \\ v &= -e_n + B^T x \geq 0, \quad y \geq 0, \end{aligned}$$

is at hand. It remains to show that in Step 2, the driving variable is always blocked, i.e., that termination with a ray is impossible.

If termination with a ray occurs, there must exist an almost complementary extreme point $(\overline{u}, \overline{v}, \overline{x}, \overline{y})$ and a vector $(\tilde{u}, \tilde{v}, \tilde{x}, \tilde{y})$ such that

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \end{bmatrix} = \begin{bmatrix} 0 & A \\ B^T & 0 \end{bmatrix} \begin{bmatrix} \tilde{x} \\ \tilde{y} \end{bmatrix}, \quad 0 \neq (\tilde{x}, \tilde{y}) \geq 0. \tag{5.10}$$

Point along the ray are of the form $(\overline{u}+\lambda\tilde{u}, \overline{v}+\lambda\tilde{v}, \overline{x}+\lambda\tilde{x}, \overline{y}+\lambda\tilde{y})$ where $\lambda \geq 0$, and for all such λ

$$(\overline{u}+\lambda\tilde{u})_i(\overline{x}+\lambda\tilde{x})_i = 0 \quad i \neq k,$$
$$(\overline{v}+\lambda\tilde{v})_i(\overline{y}+\lambda\tilde{y})_i = 0 \quad i \neq k.$$

This implies that for all $i \neq k$

$$\overline{u}_i\overline{x}_i = \tilde{u}_i\overline{x}_i = \overline{u}_i\tilde{x}_i = \tilde{u}_i\tilde{x}_i = 0,$$
$$\overline{v}_i\overline{y}_i = \tilde{v}_i\overline{y}_i = \overline{v}_i\tilde{y}_i = \tilde{v}_i\tilde{y}_i = 0.$$

It must be the case that either $\tilde{x} \neq 0$ or $\tilde{y} \neq 0$. If $\tilde{x} \neq 0$, then $\tilde{v} = B^T\tilde{x} > 0$. This implies that $\overline{y}_j + \lambda\overline{y}_j = 0$ for all j and all $\lambda \geq 0$. But then $\overline{u}+\lambda\tilde{u} < 0$, which is a contradiction. If $\tilde{x} = 0$, then (5.10) implies $\tilde{y} \neq 0$ from which it follows that $\tilde{u} = A\tilde{y} > 0$. This implies that $\overline{x}_i = 0$ for all $i \neq k$. From $\tilde{x} = 0$ it follows that $\tilde{v} = B^T\tilde{x} = 0$. Accordingly, $\tilde{v}$ must be the same vector as the one defined in Step 1 where the initial value of x_k was specified, i.e., the smallest positive value of x_k so that $-1 + m_{jk}x_k \geq 0$, where m_{jk} denotes an element in M.

By the nondegeneracy assumption, only $\overline{v}_s = 0$. The other components of $\overline{v}$ must be positive. Thus

$$\overline{y}_j + \lambda\tilde{y}_j = 0, \quad \text{for all } j \neq s.$$

We now see that the terminating ray is the original ray. This means that the almost complementary path must have returned to a previously visited extreme point which is impossible. This contradiction completes the proof. □

We now return to the example problem (5.4) in section 5.2, which is originally from [1, p. 287]. Applying the Lemke-Howson algorithm to the example problem, the solutions $\overline{z} = (\frac{1}{10}, 0, \frac{1}{10}, 0)$ and $\overline{z} = (0, \frac{1}{15}, 0, \frac{1}{15})$ are obtained. The first solution represents the situation $x = (1, 0)$ and $y = (1, 0)$ and the second represents the situation when $x = (0, 1)$ and $y = (0, 1)$. In both cases both players adopt pure strategies, i.e. they choose the same action every time.

Which one of the two solutions that will be found by the Lemke-Howson method depends on which column that is pivoted first in Step 0. There is no guarantee that the Lemke-Howson method will find all equilibrium points to a bimatrix game. In the example problem (5.4) there is an additional solution which the Lemke-Howson method cannot reach, namely $\overline{z} = (\frac{1}{90}, \frac{2}{45}, \frac{1}{90}, \frac{2}{45})$, i.e. the strategy pair $x = (\frac{1}{5}, \frac{4}{5})$ and $y = (\frac{1}{5}, \frac{4}{5})$.

6 Summary

In the report, an overview of the Linear Complementarity Problem (LCP) as an special case of the Mathematical Programming with Equilibrium Constraints (MPEC) has been given. MPEC was described as an optimization problem "in which the essential constraints are defined by a parametric variational inequality or complementarity system". The linear complementarity problem is a special case of the MPEC. Two methods for solving LCPs have been described: The interior point method and Lemke's method. The interior point method runs in polynomial time while the running time of Lemke's method is exponential. Two versions of Lemke's method were considered, one for augumented problems and one streamlined version. Stackelberg games were described as an application source for the MPEC. The bimatrix games were looked at closer as an application for the LCP. Also, the Lemke-Howson method was reviewed as a method for solving bimatrix games.

References

[1] Richard W. Cottle, Jong-Shi Pang, and Richard E. Stone. *The Linear Complementarity Problem*. Academic Press, San Diego, 1992.

[2] George B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, Princeton, New Jersey, 1963.

[3] Francisco Facchinei and Jong-Shi Pang. *Finite-Dimensional Variational Inequalities and Complementarity Problems, Volume I*. Springer-Verlag, 2000.

[4] C.A Floudas and P.M. Pardalos. *Encyclopedia of Optimization*. Kluwer Academic Press, Dordrecht, 2001.

[5] Osman Güler. Generalized linear complementarity problems. *Mathematics of Operation Research*, 20:441–448, 1995.

[6] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4:373–395, 1984.

[7] Masakazu Kojima, Shinhi Mizuno, and Akiko Yoshise. A polynomial-time algorithm for a class of linear complementarity problems. *Mathematical Programming*, 44:1–26, 1988.

[8] C. E. Lemke. Bimatrix equilibrium points and mathematical programming. *Management Science*, 11:681–689, 1965.

[9] Sven Leyffer. Mathematical programs with complementarity constraints. *SIAG/OPT Views-and-News*, 14:15–18, 2003.

[10] Zhi-Quan Luo, Jong-Shi Pang, and Daniel Ralph. *Mathematical Programs with Equilibrium Constraints*. Cambridge University Press, 1996.

[11] Xiaoyan Zhang and Puyan Nie. A Note on Stackelberg Games and Nash Games, 2006.