Path-Following for Bimatrix Games Using Lemke's Algorithm

Bernhard von Stengel

April 29, 2023

Abstract

We study three path-following methods that find at least one equilibrium of a bimatrix game, and show how to implement them as special cases of Lemke's algorithm. These methods are the global Newton method, with the Lemke-Howson algorithm as a special case, and the van den Elzen-Talman linear tracing procedure.

1 Linear complementarity

All vectors are column vectors, and $\mathbf{0}$ and $\mathbf{1}$ are the all-zero and all-one vector, respectively, their dimension depending on the context. Vectors and scalars are treated as matrices and positioned such that matrix multiplication works (as in column vector times a scalar, but scalar times a row vector).

The dimension of an LCP is denoted by N rather than n because we consider $m \times n$ bimatrix games.

The standard linear complementarity problem or LCP (q, M) is specified by an N-vector q and an $N \times N$ matrix M and states: find an N-vector z such that

$$z \geq \mathbf{0} \quad \perp \quad q + Mz \geq \mathbf{0} \tag{1}$$

where the inequalities are meant to hold and the symbol $\perp$ denotes orthogonality and thus complementarity of the respective slacks; that is, for general N-vectors a, b, c, d:

$$a \geq b \quad \perp \quad c \geq d \qquad :\Leftrightarrow \qquad a \geq b, \;\; c \geq d, \;\; (a-b)^\top (c-d) = 0\,. \tag{2}$$

In an $m \times n$ bimatrix game (A, B), let X and Y be the mixed-strategy sets of the row and column player, respectively,

$$X = \{x \in \mathbb{R}^m \mid x \geq \mathbf{0},\; \mathbf{1}^\top x = 1\}, \qquad Y = \{y \in \mathbb{R}^n \mid y \geq \mathbf{0},\; \mathbf{1}^\top y = 1\}. \tag{3}$$

A mixed Nash equilibrium or just *equilibrium* of (A, B) is a pair $(x, y) \in X \times Y$ such that there are reals u and v with

$$x \geq \mathbf{0} \quad \perp \quad Ay \leq \mathbf{1}u\,, \qquad y \geq \mathbf{0} \quad \perp \quad B^\top x \leq \mathbf{1}v\,, \tag{4}$$

where u and v are the equilibrium payoffs to the row and column player, respectively. Condition (4) is the best-response condition that every pure strategy that has positive probability must be a pure best response against the other player's mixed strategy (von Stengel, 2022, proposition 6.1).

This is a special case of an LCP, and the algorithm by Lemke (1965) for solving an LCP can be used to represent several path-following methods for finding an equilibrium of a bimatrix game. In order to do so, we will employ several variable transformations.

First, we represent the unrestricted variables u and v in (4) as differences of nonnegative variables in the form

$$u = u^+ - u^-, \quad u^+ \geq 0,\ u^- \geq 0, \qquad v = v^+ - v^-, \quad v^+ \geq 0,\ v^- \geq 0, \tag{5}$$

and the equations $\mathbf{1}^\top x = 1$ and $\mathbf{1}^\top y = 1$ as two inequalities, as follows:

$$\begin{array}{rcccccccl}
x \geq \mathbf{0} & \perp & & -Ay & +\mathbf{1}u^+ - \mathbf{1}u^- & & \geq & \mathbf{0} \\
y \geq \mathbf{0} & \perp & -B^\top x & & & +\mathbf{1}v^+ - \mathbf{1}v^- & \geq & \mathbf{0} \\
u^+ \geq 0 & \perp & -\mathbf{1}^\top x & & & & \geq & -1 \\
u^- \geq 0 & \perp & \mathbf{1}^\top x & & & & \geq & 1 \\
v^+ \geq 0 & \perp & & -\mathbf{1}^\top y & & & \geq & -1 \\
v^- \geq 0 & \perp & & \mathbf{1}^\top y & & & \geq & 1
\end{array} \tag{6}$$

Suppose that neither equilibrium payoff u or v is zero. For example, if $v < 0$ then $v^- > 0$ and the last inequality in (6) is tight, that is, $\mathbf{1}^\top y = 1$ and thus $y \in Y$. We have chosen the complementarity conditions in (6) so that the constraint matrix M is skew-symmetric ($M = -M^\top$) if the game is zero-sum ($B = -A$). It will be convenient to have an LCP matrix M where the sub-matrices $-A$ and $-B^\top$ are positive, that is,

$$A < 0, \qquad B < 0, \tag{7}$$

which can always be achieved by subtracting a sufficiently large constant from the entries of A and B. Then any equilibrium payoffs u and v are negative, so that we can choose

$$u^+ = v^+ = 0, \qquad u = -u^-, \qquad v = -v^-, \tag{8}$$

and (6) simplifies to

$$\begin{array}{rcccccl}
x \geq \mathbf{0} & \perp & & -Ay & -\mathbf{1}u^- & & \geq \mathbf{0} \\
y \geq \mathbf{0} & \perp & -B^\top x & & & -\mathbf{1}v^- & \geq \mathbf{0} \\
u^- \geq 0 & \perp & \mathbf{1}^\top x & & & & \geq 1 \\
v^- \geq 0 & \perp & & \mathbf{1}^\top y & & & \geq 1\ .
\end{array} \tag{9}$$

By (7) and complementarity, $u^- = -u > 0$ and $v^- = -v > 0$ in any solution (x, y, u^-, v^-) to (9), and the last two inequalities are therefore tight, that is, $x \in X$ and $y \in Y$.

A second variable transformation changes the "homotopy parameter" λ in the "global Newton method" (see Section 3). We first summarize Lemke's algorithm.

2 Lemke's algorithm

Lemke (1965) described an algorithm for solving the LCP (1). It uses an additional N-vector d, called *covering vector*, with a corresponding scalar variable z_0, and computes with *basic solutions* to the augmented system

$$z \geq \mathbf{0}, \quad z_0 \geq 0, \qquad w = q + Mz + dz_0 \geq \mathbf{0}, \qquad z^\top w = 0\,. \tag{10}$$

Any solution z, z_0 to (10) is called *almost complementary*, and *complementary* if $z_0 = 0$, which implies (1). The vector d is assumed to fulfill

$$d \geq \mathbf{0}, \qquad q_i < 0 \quad \Rightarrow \quad d_i > 0 \quad (1 \leq i \leq N). \tag{11}$$

This implies that for $z = \mathbf{0}$ and all sufficiently large z_0 we have $w = q + dz_0 \geq \mathbf{0}$ and $z^\top w = 0$, so that (10) holds. The set of these almost complementary solutions is called the *primary ray*.

For initialization, let z_0 be minimal such that $w = q + dz_0 \geq \mathbf{0}$. Unless $q \geq \mathbf{0}$ (in which case the LCP is solved immediately), z_0 is positive and some component w_i of w is zero. A first pivoting step lets z_0 enter and w_i leave the basis. The algorithm then performs a sequence of *complementary pivoting* steps. At each step, one variable of a complementary pair (z_i, w_i) leaves and then its *complement* (the other variable) enters the basis, which maintains the condition $z^\top w = 0$. With a suitable lexicographic symbolic perturbation, the leaving variable (including the first leaving variable w_i when z_0 enters) is always unique, so that the algorithm follows a unique path. The goal is that eventually z_0 leaves the basis and then has value zero, so that the LCP (1) is solved.

Lemke's algorithm could also terminate with a *secondary ray*, which happens when the entering variable can assume arbitrarily positive values (as in the simplex algorithm for linear programming for an unbounded objective function). Certain conditions on the LCP data (q, M) exclude this ray termination. These hold if A and B are negative as in (7), as shown in Koller, Megiddo, and von Stengel (1996, theorem 4.1) if the LCP is presented in its original form (6), and also in its shortened form (9) using additional considerations. We will be able to exclude ray termination with a more direct argument.

3 Path-following of equilibria

The general "homotopy" approach to finding an equilibrium of a game in strategic form is to modify the game to a one-parameter set of games with a real parameter λ such that $\lambda = 0$ corresponds to the given game, and a sufficiently large value of λ has an equilibrium that is easy to find, which serves as a *starting point* of the algorithm. Subsequently, λ is lowered (with possible intermittent increases) while "tracing" the equilibrium of the parameterized game, until $\lambda = 0$ and an equilibrium of the given game is found. The result is a path of parameterized games and a corresponding equilibrium for each game. For a bimatrix game (A, B) and the methods considered here, the path is piecewise linear. Up to scaling, it consists of line segments in the set $X \times Y$ of mixed-strategy pairs, represented by the pivoting steps of Lemke's algorithm.

For brevity, we first summarize the different methods and how to represent them with Lemke's algorithm, with details in the following subsections. We always assume negative payoff matrices A and B as in (7) and consider the LCP (1) as in (9) with $N = m + n + 2$ and

$$z = \begin{bmatrix} x \\ y \\ u^- \\ v^- \end{bmatrix}, \qquad q = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ -1 \\ -1 \end{bmatrix}, \qquad M = \begin{bmatrix} & -A & \mathbf{1} & \\ -B^\top & & & -\mathbf{1} \\ \mathbf{1}^\top & & & \\ & \mathbf{1}^\top & & \end{bmatrix}. \tag{12}$$

In the augmented LCP (10), the covering vector d is chosen according to the different methods.

- The *global Newton method*, with subsidy vectors $a \in \mathbb{R}^m$ and $b \in \mathbb{R}^n$ for the rows and columns, traces equilibria of

$$\Gamma_\lambda = (A + a\lambda\mathbf{1}^\top,\ B + \mathbf{1}\lambda b^\top)\,. \tag{13}$$

  This is represented in Lemke's algorithm with

$$d = \begin{bmatrix} \mathbf{1}\alpha - a \\ \mathbf{1}\beta - b \\ 1 \\ 1 \end{bmatrix}, \qquad z_0 = \frac{\lambda}{1+\lambda} = \frac{1}{1/\lambda + 1}, \qquad \lambda = \frac{z_0}{1 - z_0} \tag{14}$$

  with sufficiently large constants α and β such that $\mathbf{1}\alpha - a$ and $\mathbf{1}\beta - b$ are positive. The parameters $\lambda \in [0, \infty]$ and $z_0 \in [0, 1]$ are in a monotonic bijection via (14).

- The *Lemke-Howson (LH)* method uses a *missing label* k in $\{1, \ldots, m + n\}$ for a row or column strategy that is is subsidized. This is represented with $d \in \mathbb{R}^N$ defined by

$$d_i = 1 \quad (1 \le i \le N,\ i \ne k), \qquad d_k = 0\,. \tag{15}$$

This has some disadvantages compared to the standard implementation of the LH method, as will be discussed, but maps nicely to Lemke's algorithm.

- The *tracing procedure* by van den Elzen and Talman uses a *prior* or starting point $(\bar{x}, \bar{y}) \in X \times Y$. It traces equilibria $(\hat{x}, \hat{y})$ of a game where the players play with weight (or probability) $\tau \in [0, 1]$ against the prior and with weight $1 - \tau$ against their actual strategies $(\hat{x}, \hat{y})$, starting with $\tau = 1$ and ending with $\tau = 0$. This is represented with

$$d = \begin{bmatrix} -A\bar{y} \\ -B^\top \bar{x} \\ \mathbf{1} \\ \mathbf{1} \end{bmatrix}, \qquad z_0 = \tau, \qquad x = \hat{x}(1 - z_0), \quad y = \hat{y}(1 - z_0) \tag{16}$$

with $z = (x, y, u^-, v^-)^\top$ as in (12).

We now consider these methods in detail.

3.1 The global Newton method

The global Newton method (Govindan and Wilson, 2003) works for any finite game in strategic form, but we consider it here for two players for an $m \times n$ bimatrix game (A, B). Let $a \in \mathbb{R}^m$ and $b \in \mathbb{R}^n$ where a_i and b_j are considered a "subsidy" (or penalty if negative) for row i and column j, respectively. These subsidies are scaled simultaneously with λ and added to each respective row and column, which defines the parameterized game (13).

If a and b have a unique maximum a_i and b_j, say, then for sufficiently large λ the game Γ_λ will have the pure-strategy pair (i, j) as its unique equilibrium as the starting point. Lowering λ will then either lead to $\lambda = 0$ and having that same equilibrium as an equilibrium of (A, B), or for some $\bar{\lambda} > 0$ give a new pure strategy k of either player as a new best response. At that point, the game $\Gamma_{\bar{\lambda}}$ is necessarily degenerate, but has generically only a line segment, at this initial step some convex combinations of i and k (if k belongs to the row player), as an equilibrium component. Traversing that equilibrium component in $\Gamma_{\bar{\lambda}}$ then may either lead to a new best response, or dropping a previous best response (like i), which allows to change λ again (normally a decrease, but possibly also an increase of λ).

Generically, the encountered equilibrium components are one-dimensional, and no bifurcation occurs; when implemented with Lemke's algorithm, lexicographic degeneracy resolution creates a unique path, even if the maximum entries of a and b are not unique. The computed path consists of games Γ_λ, each with an associated unique equilibrium, which in case of intermittent increases of λ depends on the path history. The path terminates with $\lambda = 0$ because for large λ the game Γ_λ has a unique equilibrium, so the path cannot continue with λ becoming infinite (which would be a secondary ray in Lemke's algorithm).

We now explain (14). The constants α and β are just added to all entries of A and B, respectively, and do not change best responses. For the game Γ_λ in (13), the conditions for an equilibrium (x, y) state

$$\begin{array}{rclllll} x \geq \mathbf{0} & \perp & & -Ay & -\mathbf{1}u^- & & -a\lambda \geq \mathbf{0} \\ y \geq \mathbf{0} & \perp & -B^\top x & & & -\mathbf{1}v^- & -b\lambda \geq \mathbf{0} \\ u^- \geq 0 & \perp & \mathbf{1}^\top x & & & & \geq 1 \\ v^- \geq 0 & \perp & & \mathbf{1}^\top y & & & \geq 1\,. \end{array} \tag{17}$$

Assuming that sufficiently large constants $\mathbf{1}\alpha$ and $\mathbf{1}\beta$ are added to $-a$ and $-b$, this would work with Lemke's algorithm with $z_0 = \lambda$ and $d = (\mathbf{1}\alpha - a, \mathbf{1}\beta - b, 0, 0)^\top$ if u^- and v^- could be pivoted into the basis before z_0. However, this is not a standard initialization where z_0 enters first, which requires condition (11) to hold, which fails here because $q = (\mathbf{0}, \mathbf{0}, -1, -1)^\top$. Instead, we divide all $m + n$ inequalities in (17) by $1 + \lambda$, which is always positive, and let

$$\hat{x} = x\tfrac{1}{1+\lambda}, \quad \hat{y} = y\tfrac{1}{1+\lambda}, \quad \hat{u} = u^-\tfrac{1}{1+\lambda}, \quad \hat{v} = v^-\tfrac{1}{1+\lambda}, \quad \tau = \tfrac{\lambda}{1+\lambda}, \quad 1 - \tau = \tfrac{1}{1+\lambda}, \tag{18}$$

so that $\mathbf{1}x \geq 1$ becomes $\mathbf{1}\hat{x} \geq 1 - \tau$ and $\mathbf{1}y \geq 1$ becomes $\mathbf{1}\hat{y} \geq 1 - \tau$, and (17) becomes

$$\begin{array}{rclllll} \hat{x} \geq \mathbf{0} & \perp & & -A\hat{y} & -\mathbf{1}\hat{u} & & -\ a\tau \geq \mathbf{0} \\ \hat{y} \geq \mathbf{0} & \perp & -B^\top \hat{x} & & & -\mathbf{1}\hat{v} & -\ b\tau \geq \mathbf{0} \\ \hat{u} \geq 0 & \perp & \mathbf{1}^\top \hat{x} & & & & +\ \ \tau \geq 1 \\ \hat{v} \geq 0 & \perp & & \mathbf{1}^\top \hat{y} & & & +\ \ \tau \geq 1\,. \end{array} \tag{19}$$

With $-a$ replaced by $\mathbf{1}\alpha - a$, and $-b$ replaced by $\mathbf{1}\beta - b$, and $z_0 = \tau$, this is the system (14) apart from the naming of the variables $\hat{x}, \hat{y}, \hat{u}, \hat{v}$, with the standard initialization of Lemke's algorithm. The variable z_0 enters the basis with value $z_0 = 1$, which corresponds to $\lambda = \infty$. The resulting basis is degenerate, and lexicographic perturbation implies that $\hat{v}$ enters the basis next, and $\hat{u}$ also has to enter before z_0 can be reduced. However, the resulting path is unique and cannot terminate with $z_0 = 1$ (or $\lambda = \infty$) because this would mean an alternative way to start (a secondary ray is different from the primary ray).

During the initialization phase when $z_0 = 1$, we have $\hat{x} = \mathbf{0}$ and $\hat{y} = \mathbf{0}$, but during the main computation when $z_0 < 1$, the pair $(\hat{x}, \hat{y})$ multiplied by $\frac{1}{1-z_0}$ (that is, by $1 + \lambda$) is the equilibrium of the parameterized game Γ_λ. When $z_0 = 0$ and the algorithm terminates, then $(\hat{x}, \hat{y})$ is an (unscaled) mixed equilibrium of the original game Γ_0.

3.2 The Lemke-Howson algorithm

The algorithm by Lemke and Howson (1964) has a nice visualization due to Shapley (1974) (see also von Stengel, 2022, chapter 9). We number the $m+n$ pairs of inequalities in (4) with $1,\ldots,m$ for the m pure strategies of the row player, and with $m+1,\ldots,m+n$ for the n pure strategies of the column player. Each number in $\{1,\ldots,m+n\}$ is called a *label*, and any (x,y,u,v) in $X\times Y\times\mathbb{R}\times\mathbb{R}$ is said to have label k if one of the corresponding inequalities is tight; we also say (x,y) has such a label if that is the case for suitable u and v. That is, (x,y) has label i in $\{1,\ldots,m\}$ if $x_i=0$ or $(Ay)_i=u$, assuming $Ay\le\mathbf{1}u$, and has label j in $\{m+1,\ldots,m+n\}$ if $y_j=0$ or $(B^\top x)_j=v$, assuming $B^\top x\le\mathbf{1}v$. In other words, (x,y) has a pure strategy of either player as a label if that pure strategy is played with probability zero or is a best response. An equilibrium (x,y) is characterized by having all labels $1,\ldots,m+n$.

The Lemke-Howson algorithm generalizes the equilibrium condition by allowing one label, say k, to be *missing*, and considers all (x,y,u,v) in $X\times Y\times\mathbb{R}\times\mathbb{R}$ such that $Ay\le\mathbf{1}u$ and $B^\top x\le\mathbf{1}v$ that have at least the labels in $\{1,\ldots,m+n\}-\{k\}$. These tuples are called *k-almost complementary*, and for a nondegenerate game define a collection of line segments and one ray that form a set of paths and cycles. The endpoints of the paths are the equilibria of (A,B). The ray is given by the pure strategy k and its (pure) best response, where in the system (4) the constraints $Ay\le\mathbf{1}u$ for the payoff variable u if $k\in\{1,\ldots,m\}$, or the constraints $B^\top x\le\mathbf{1}v$ for the payoff variable v if $k\in\{m+1,\ldots,m+n\}$, need not be tight because k is not required to be a best response. This works for the initial solutions given by the ray because k is the only pure strategy of the respective player that has positive probability, and the other player uses the (generically unique) pure best response to k; hence, all labels except k are present.

The Lemke-Howson algorithm is a special case of the global Newton method by setting the pair (a,b) of "subsidies" in (13) to be equal to the kth unit vector e_k in $\mathbb{R}^{m+n}$ (that is, the kth component of e_k is 1 and all other components are 0). That way, the pure strategy k is "subsidized" via λ so that it is a best response in Γ_λ (and all other pure strategies are in equilibrium) but not in the original game (A,B). The algorithm ends with $\lambda=0$ because either k has probability zero or becomes a best response in Γ_0, that is, in (A,B).

With (a,b) as the unit vector e_k for the missing label k, we can choose $\alpha=\beta=1$ in (14), which gives (15). When running Lemke's algorithm this way, it indeed emulates the LH method, where during the LCP computation the LCP variables x and y in (12) have their components sum to $1-z_0$ and need to be re-scaled to represent the steps of the LH algorithm. This is not possible in an initial phase where $z_0=1$ (and thus $x=\mathbf{0}$ and $y=\mathbf{0}$, even though some components of x

and y may already be basic. This initial phase requires a lot (typically $m + n$) degenerate pivots because the vector d together with the column entries for u^- and v^- generates a large number of basic variables with value zero. As long as these are potential leaving variables, these need to be pivoted out before z_0 can shrink in value.

A standard implementation of the LH algorithm uses two best-response polytopes $P = \{x \in \mathbb{R}^m \mid x \geq \mathbf{0},\ B^\top x \leq \mathbf{1}\}$ and $Q = \{y \in \mathbb{R}^n \mid Ay \leq \mathbf{1},\ y \geq \mathbf{0}\}$ with positive payoff matrices A and B (see von Stengel, 2022, section 9.6f), with $(\mathbf{0}, \mathbf{0})$ as an initial "artificial" equilibrium that is completely labeled. Then pivoting alternates between two disjoint systems, one for x and one for y, which means that the LCP tableau is always only half full. In the implementation described here, the tableau is during the intermittent computations three-quarters full. It has, however, the advantage that only the LCP parameters for the standard algorithm by Lemke have to be adapted, not the algorithm itself.

3.3 The van den Elzen-Talman linear tracing procedure

The algorithm by van den Elzen and Talman (1999) is another path-following method of "tracing" equilibria in a certain restricted game, which is defined via a "prior" or starting point $(\bar{x}, \bar{y})$ in the mixed-strategy space $X \times Y$. For $\tau \in [0, 1]$, consider the restricted set of strategy profiles

$$\begin{aligned} S(\bar{x}, \bar{y}, \tau) &= \{\, (x, y) \in X \times Y \mid x \geq \bar{x}\tau,\ y \geq \bar{y}\tau \} \\ &= \{\, (\hat{x} + \bar{x}\tau, \hat{y} + \bar{y}\tau) \mid \mathbf{1}^\top \hat{x} = 1 - \tau,\ \hat{x} \geq \mathbf{0},\ \mathbf{1}^\top \hat{y} = 1 - \tau,\ \hat{y} \geq \mathbf{0}\} \\ &= \{\, (\tilde{x}(1 - \tau) + \bar{x}\tau, \tilde{y}(1 - \tau) + \bar{y}\tau) \mid (\tilde{x}, \tilde{y}) \in X \times Y \}\,. \end{aligned} \tag{20}$$

For (x, y) in the set $S(\bar{x}, \bar{y}, \tau)$, every mixed-strategy probability x_i or y_j has to be at least $\bar{x}_i$ or $\bar{y}_j$, respectively. For $\tau = 1$, clearly $S(\bar{x}, \bar{y}, 1) = \{(\bar{x}, \bar{y})\}$. In the second equation in (20), $\hat{x} = x - \bar{x}\tau$ and $\hat{y} = y - \bar{y}\tau$ (we will use $\hat{x}$ and $\hat{y}$ as LCP variables), and in the last equation of (20) the mixed strategies $\tilde{x}$ and $\tilde{y}$ are arbitrary for $\tau = 1$ and uniquely defined by $\tilde{x} = x\frac{1}{1-\tau}$ and $\tilde{y} = y\frac{1}{1-\tau}$ if $0 \leq \tau < 1$.

The "linear tracing procedure" of van den Elzen and Talman (1999) follows a path of equilibria (x, y) restricted to the strategy set $S(\bar{x}, \bar{y}, \tau)$, parameterized by τ. It starts with $\tau = 1$ at the prior $(\bar{x}, \bar{y})$, which generically has a pair $(\tilde{x}, \tilde{y})$ of pure best responses. The corresponding convex combination $(\tilde{x}(1 - \tau) + \bar{x}\tau, \tilde{y}(1 - \tau) + \bar{y}\tau)$ stays a best response against itself (i.e., is an equilibrium) for τ slightly less than 1. Subsequent lowering of τ either stops at $\tau = 0$ or for some $\bar{\tau}$ introduces a new best response of some player which is then increased, with τ staying fixed, until τ can be changed again. Throughout, the path as represented in (20) stays inside the strategy space (X, Y), and (in the generic case) will not bifurcate or return to $(\bar{x}, \bar{y})$

but will eventually terminate at $\tau = 0$ with an equilibrium of the original game (A, B).

The interpretation of the linear tracing procedure is that the prior represents a preconception of the players of what their opponent will do. The players reply first (for $\tau = 1$) only to that preconception, and then gradually, with weight $1 - \tau$, take their actual play $(\tilde{x}, \tilde{y})$ into account, by reacting to the respective convex combination with the prior which has weight τ, adjusting their actions until $\tau = 0$. Possible intermittent increases of τ in this adjustment process are possible.

Balthasar (2010) has shown, and it is easy to see, that the path of the linear tracing procedure for $0 \le \tau < 1$ is in one-to-one correspondence with the path for the global Newton method via

$$\lambda = \frac{\tau}{1-\tau} = \frac{1}{1/\tau - 1}, \qquad (a, b) = (A\bar{y}, B^\top \bar{x}) \tag{21}$$

in (13), where $\lambda \in [0, \infty)$ and $\tau \in [0, 1)$ are strictly monotonic functions of each other. The starting point $(\bar{x}, \bar{y})$ of the linear tracing procedure corresponds to the ray of equilibria of Γ_λ when $\lambda \to \infty$ as $\tau \to 1$. We have used exactly this transformation in (18) to use the parameter τ instead of λ for the global Newton method.

Balthasar (2010) has also shown that there are open sets of games where an equilibrium of positive index is *not* found via the linear tracing procedure. For example, the symmetric 3×3 game (A, B)

$$A = B^\top = \begin{bmatrix} 0 & -1 & -c \\ -c & 0 & -1 \\ -1 & -c & 0 \end{bmatrix} \tag{22}$$

has a completely mixed equilibrium, which has positive index, which for $1 < c < 2$ is not found for any generic prior $(\bar{x}, \bar{y})$. Condition (21) shows that the global Newton method has more general parameters (a, b) than the linear tracing procedure. For example, via λ in (13) the strategies of the two players may vary by different "speeds" of change for a given direction (a, b), which cannot be emulated by a suitable prior $(\bar{x}, \bar{y})$. It is open whether the completely mixed equilibrium in (22) (or in general any equilibrium of positive index) can be found with the global Newton method.

Clearly, the linear tracing procedure is implemented via Lemke's algorithm by (16), using $a = A\bar{y}$ and $b = B^\top \bar{x}$ in (19). It would be of interest if it can serve as a good substitute for an evolutionary dynamics (like the replicator dynamics). One first question in this direction is whether the completely mixed equilibrium of the game in (22) is dynamically stable in an evolutionary setting.

4 Random starting points

One way of using the linear tracing procedure is for equilibrium selection. We propose to choose starting vectors $(\bar{x}, \bar{y})$ where $\bar{x}$ and $\bar{y}$ are each chosen from a uniform distribution on their respective mixed-strategy simplices. The corresponding percentages of which equilibrium is found is then an indication of the "prevalence" of this equilibrium, in particular a "practical" uniqueness if the found equilibrium is always the same.

[Details on implementation.]

References

Balthasar, A. (2010). Equilibrium tracing in strategic-form games. *Economic Theory* 42(1), 39–54.

Govindan, S. and R. Wilson (2003). A global Newton method to compute Nash equilibria. *Journal of Economic Theory* 110(1), 65–86.

Koller, D., N. Megiddo, and B. von Stengel (1996). Efficient computation of equilibria for extensive two-person games. *Games and Economic Behavior* 14(2), 247–259.

Lemke, C. E. (1965). Bimatrix equilibrium points and mathematical programming. *Management Science* 11(7), 681–689.

Lemke, C. E. and J. T. Howson, Jr (1964). Equilibrium points of bimatrix games. *Journal of the Society for Industrial and Applied Mathematics* 12(2), 413–423.

Shapley, L. S. (1974). A note on the Lemke-Howson algorithm. *Mathematical Programming Study 1: Pivoting and Extensions*, 175–189.

van den Elzen, A. and D. Talman (1999). An algorithmic approach toward the tracing procedure for bi-matrix games. *Games and Economic Behavior* 28(1), 130–145.

von Stengel, B. (2022). *Game Theory Basics*. Cambridge University Press, Cambridge, UK.