Tracing Equilibria in Extensive Games by Complementary Pivoting

Bernhard von Stengel* Antoon van den Elzen† Dolf Talman‡

December 20, 1996

Abstract. An algorithm is presented for computing an equilibrium of an extensive two-person game with perfect recall. The equilibrium is traced on a piecewise linear path from an arbitrary starting point. If this is a pair of completely mixed strategies, then the equilibrium is normal form perfect. The normal form computation is performed efficiently using the sequence form, which has the same size as the extensive game itself.

Keywords. Efficient computation, extensive form game, lexicographic pivoting, linear complementarity, normal form perfect, sequence form, tracing procedure.

1. Introduction

Consider a two-person game in extensive form where the players have perfect recall. This paper presents an efficient algorithm for finding an equilibrium of such a game with appealing properties. Given the game, it computes a sample equilibrium that is *normal form perfect*. The algorithm generates a piecewise linear path in the strategy space. An arbitrary strategy pair is chosen as *starting point*, serving as a parameter for the computation. Various starting points can be tried to find possibly different equilibria. The probabilities for playing strategies at the starting point become relative *mistake* probabilities for play-

*B. von Stengel, Theoretische Informatik, ETH Zürich, 8092 Zürich, Switzerland. Email: stengel@inf.ethz.ch

†A. H. van den Elzen, Department of Econometrics and CentER for Economic Research, Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands. Email: elzen@kub.nl

‡A. J. J. Talman, Department of Econometrics and CentER for Economic Research, Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands. Email: talman@kub.nl

ing suboptimal strategies in mixed strategies that approximate the computed equilibrium. Thus, this equilibrium is perfect if the starting point is completely mixed.

The algorithm is highly efficient because it uses a data structure that does not take more space than the extensive game itself. This is the *sequence form* of the extensive game (Romanovsky, 1962; von Stengel, 1996). In the sequence form, mixed strategies of a player are identified in a natural way when they are realization equivalent, that is, induce the same behavior along the path of play and therefore the same payoffs. After this identification, they belong to a strategy space (a certain polytope) of low dimension, which is equal to the total number of moves of the player. In contrast, the mixed strategy simplex has typically *exponential* dimension in the size of the game tree. This exponential blowup makes standard methods applied to the normal form impractical.

The computation can be interpreted game-theoretically. In this interpretation, the starting point represents a *prior* against which the players react initially. Next, they gradually adjust their behavior by using information about the strategy that is actually played. Each point on the piecewise linear path is an equilibrium of a restricted game where the prior is played with some probability being initially one and then being decreased towards zero (with possible intermittent increases). This mimicks the *linear tracing procedure* as formulated by Harsanyi and Selten (1988), who use it to determine an equilibrium for a basic game starting from the given prior. Basic games result from the so-called standard form of an extensive game after a reduction procedure. Here we emulate, up to projection, the tracing procedure for the normal form, which can be considered as a special case of the standard form for this purpose.

The sequence form strategy space of a player is a linear projection of his mixed strategy simplex. The relevant properties of mixed strategies for the above interpretation are preserved, like lying on a line or being completely mixed. In a sense, we merely perform normal form computations efficiently. Therefore, the emulated tracing procedure and the perfection of the equilibrium apply to the normal form of the game.

Our algorithm is a synthesis of previous, partly independent work by the authors and Daphne Koller and Nimrod Megiddo. For bimatrix games, van den Elzen and Talman (1991) (see also van den Elzen, 1993) described a complementary pivoting algorithm that traces a given prior to an equilibrium. If the prior is completely mixed, the computed path leads to a perfect equilibrium. Generically, the algorithm represents the linear tracing procedure by Harsanyi and Selten up to projection (van den Elzen and Talman, 1995). Koller, Megiddo, and von Stengel (1996) applied the complementary pivoting algorithm by Lemke (1965) to the sequence form. As a special case, it can also be applied to a game in normal form. If one chooses in a specific way a certain parameter in Lemke's algorithm, the so-called covering vector, then one obtains the algorithm by van den Elzen and Talman.

Here, we derive the covering vector for Lemke's algorithm analogously from an arbitrary starting point, but applied to the sequence form. As a consequence of this initialization, the generated path stays in the compact strategy space. This simplifies the earlier proof of Koller, Megiddo, and von Stengel (1996) that the algorithm terminates, which was very technical. However, degeneracy has to be dealt with properly (using a known lexicographic pivoting rule) since it arises naturally from the game tree structure. It can easily happen that along the generated path of strategies, it becomes no longer optimal to put weight on an entire branch of the game tree. At that moment more than one variable becomes zero and degeneracy occurs. Thus, degeneracy in extensive form games is related to the structure of the game tree. For generic bimatrix games, degeneracy could be disregarded.

Charnes (1953) described the solution of "constrained" zero-sum games where each player's strategy space is a polytope. Romanovsky (1962) derived from an extensive game such a constrained matrix game which is equivalent to the sequence form. However, this Russian publication was overlooked in the English-speaking community. Eaves (1973) applied Lemke's algorithm to games which include polyhedrally constrained bimatrix games, but with different parameters than we do. Dai and Talman (1993) described an algorithm that corresponds to ours but requires simple polyhedra as strategy spaces, which is not the case for the sequence form. Wilson (1972) described a method for solving extensive two-person games, where best responses, which serve as pivoting columns for the Lemke–Howson algorithm, are generated directly from the game tree. This algorithm is based on the normal form and efficient only in the sense that it uses few pure strategies with positive probability, a claim made more precise by Koller and Megiddo (1996). Except for very small game trees, the sequence form has smaller size than any normal form representation, among other things because of its sparse payoff matrix. Wilson (1992) adapted the Lemke–Howson algorithm for computing a "simply stable" equilibrium of a bimatrix game. The computed equilibrium is also perfect. The algorithm uses a lexicographic perturbation technique implying that the mistake probabilities for pure strategies in approximating mixed strategies have different orders of magnitude, according to an initially chosen order of the pure strategies. In contrast, we can "fine-tune" mistake probabilities with the choice of the starting point. It is open if Wilson's algorithm for finding simply stable equilibria can be usefully applied to the sequence form. Recent surveys on algorithms for computing Nash equilibria are McKelvey and McLennan (1996) and von Stengel (1997).

The setup of the paper is as follows. In Section 2 we recall the notion of the sequence form and its derivation from the extensive form game, with particular emphasis on the geometry of the strategy spaces. In Section 3 we consider optimal play in the sequence form. Section 4 is devoted to Lemke's algorithm, adapted for the particular covering vector. The path computed by the algorithm is illustrated in Section 5. We elaborate on the game-theoretic interpretation in Section 6. In the Appendix, we discuss the treatment of degeneracy.

2. Sequence form strategy spaces

We consider extensive two-person games, with conventions similar to von Stengel (1996) and Koller, Megiddo, and von Stengel (1996). An extensive game is given by a finite tree, payoffs at the leaves, chance moves (with positive probabilities), and information sets partitioning the set of decision nodes. The *choices* of a player are denoted by labels of tree edges. For simplicity, labels corresponding to different choices (anywhere in the tree) are distinct. For a particular player, any node of the tree defines a *sequence* of choices given by the respective labels (for his or her moves only) on the path from the root to the node. We assume that both players have *perfect recall*. By definition, this means that all nodes in an information set h of a player define for him (or her) the same sequence σ_h of choices. Under that assumption, each choice c at h is the last choice of a unique sequence $\sigma_h c$. This defines all possible sequences of a player except for the empty sequence $\emptyset$. The set of choices at an information set h is denoted C_h. The set of information sets of player i is H_i, and the set of his sequences is S_i, so

$$S_i = \{\emptyset\} \cup \{\sigma_h c \mid h \in H_i,\, c \in C_h\}.$$

This implies that the number of sequences of player i, apart from the empty sequence, is equal to his total number of moves, that is, $|S_i| = 1 + \sum_{h \in H_i} |C_h|$. This number is linear in the size of the game tree.

A *behavior strategy* β of player i is given by probabilities $\beta(c)$ for his choices c which fulfill $\beta(c) \geq 0$ and $\sum_{c \in C_h} \beta(c) = 1$ for all h in H_i. This definition of β can be extended to the sequences σ in S_i by writing

$$\beta[\sigma] = \prod_{c \text{ in } \sigma} \beta(c). \tag{2.1}$$

A *pure strategy* π is a behavior strategy with $\pi(c) \in \{0,1\}$ for all choices c. The set of pure strategies of player i is denoted P_i. Thus, $\pi[\sigma] \in \{0,1\}$ for all sequences σ in S_i. The pure strategies π with $\pi[\sigma] = 1$ are those "agreeing" with σ by prescribing all the choices in σ (and arbitrary choices at the information sets not touched by σ).

In the *normal form* of the extensive game, one considers pure strategies and their probability mixtures. A *mixed strategy* μ of player i assigns a probability $\mu(\pi)$ to every π in P_i. In the *sequence form* of the extensive game, one considers the sequences of a player instead of his pure strategies. A randomized strategy of player i is described by the *realization probabilities* of playing the sequences σ in S_i. For a behavior strategy β, these are obviously $\beta[\sigma]$ as in (2.1). For a mixed strategy μ of player i, they are given by

$$\mu[\sigma] = \sum_{\pi \in P_i} \pi[\sigma]\mu(\pi). \tag{2.2}$$

For player 1, this defines a map x from S_1 to $\mathbb{R}$ by $x(\sigma) = \mu[\sigma]$ for σ in S_1 which we call the *realization plan* of μ or a realization plan for player 1. A realization plan for player 2, similarly defined on S_2, is denoted y. Realization plans have the following important properties (Koller and Megiddo, 1992; von Stengel, 1996).

Lemma 2.1. *For player 1, x is the realization plan of a mixed strategy iff $x(\sigma) \geq 0$ for all $\sigma \in S_1$ and*

$$
\begin{aligned}
x(\emptyset) &= 1, \\
\sum_{c \in C_h} x(\sigma_h c) &= x(\sigma_h) \qquad \textit{for all } h \in H_1.
\end{aligned}
\tag{2.3}
$$

A realization plan y of player 2 is characterized analogously.

Lemma 2.2. *Two mixed strategies μ and μ' of player i are realization equivalent iff they have the same realization plan, that is, iff $\mu[\sigma] = \mu'[\sigma]$ for all $\sigma \in S_i$.*

The equations (2.3) hold for the realization probabilities $x(\sigma) = \beta[\sigma]$ for a behavior strategy β and thus for every pure strategy π, and therefore for their convex combinations in (2.2) with the probabilities $\mu(\pi)$.

For Lemma 2.2, equation (2.2) should be regarded the other way, defining a linear map from $\mathbb{R}^{|P_i|}$ to $\mathbb{R}^{|S_i|}$ that maps $(\mu(\pi))_{\pi \in P_i}$ to $(\mu[\sigma])_{\sigma \in S_i}$ with the fixed coefficients $\pi[\sigma]$, $\pi \in P_i$. Mixed strategies with the same image under this map define the same realization probabilities for all nodes of the tree irrespective of the strategy of the other player, as stated in Lemma 2.2. The simplex of mixed strategies is thereby mapped to the polytope defined by the linear constraints in Lemma 2.1. The vertices of this polytope are the realization plans of pure strategies. These are unique except for identifying realization equivalent pure strategies (as in the *reduced normal form* for generic payoffs, here RNF for short). These vertices may be exponential in number like in the mixed strategy simplex, but the dimension of the polytope is much smaller since it is linear in the size of the game tree. For player i, this dimension is the number $|S_i|$ of variables minus the number $1 + |H_i|$ of equations (2.3) (which are linearly independent), so it is $\sum_{h \in H_i}(|C_h| - 1)$. Because of this reduction in dimension, mixed strategies are "projected" to realization plans, but without losing any relevant strategic information in the extensive game.

In the sequence form, the *strategy space* of each player is the polytope of his realization plans. A line in this strategy space is the image of a line in the mixed strategy simplex, which is usually not unique, however. Any realization plan x of player 1 (and similarly y for player 2) naturally defines a behavior strategy β where the probability for making the move c is $\beta(c) = x(\sigma_h c)/x(\sigma_h)$ (which is arbitrary — for example, $\beta(c) = 1/|C_h|$ — if $x(\sigma_h) = 0$, that is, the information set h is irrelevant). However, this is obviously not a linear "inverse map" if β is regarded as a special mixed strategy.

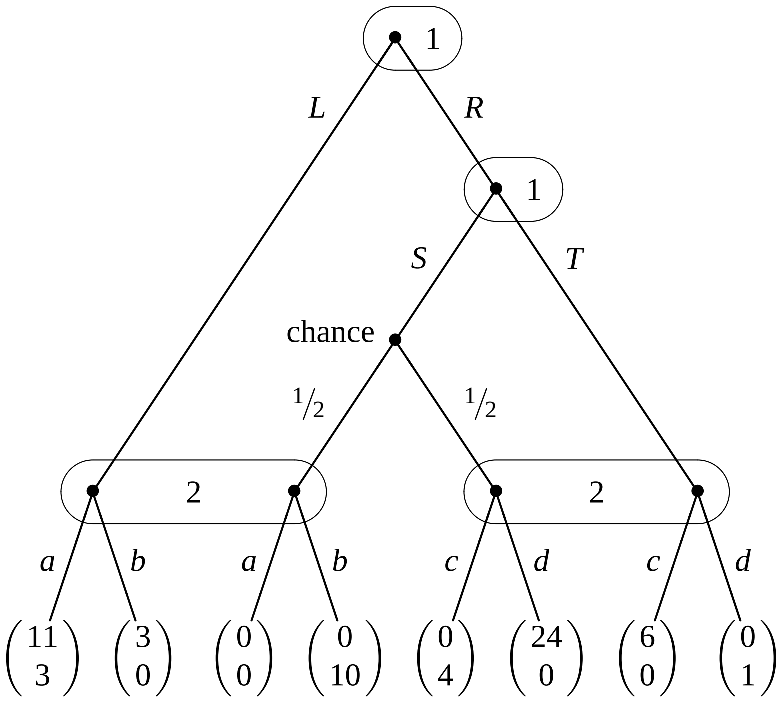

Figure 2.1. A two-person extensive game.

Figure 2.1 shows an extensive game where the choices of player 1 and player 2 are denoted by the upper and lower case letters L,R,S,T and a,b,c,d, respectively. The payoff vectors are listed at the bottom with the first and second component representing the payoff to player 1 and 2, respectively. The sets of sequences are $S_1 = \{\emptyset, L, R, RS, RT\}$ and $S_2 = \{\emptyset, a, b, c, d\}$. We consider realization plans as vectors $x = (x_\sigma)_{\sigma \in S_1}$ and $y = (y_\sigma)_{\sigma \in S_2}$, here both with five components; the sequences are written as subscripts unless they are complicated expressions like in (2.3). According to Lemma 2.1, these vectors are characterized by

$$x \geq \mathbf{0}, \quad Ex = e \qquad \text{and} \qquad y \geq \mathbf{0}, \quad Fy = f \tag{2.4}$$

(the inequalities hold componentwise, and **0** denotes a vector or matrix of zeroes), with

$$E = \begin{pmatrix} 1 & & & & \\ -1 & 1 & 1 & & \\ & & -1 & 1 & 1 \end{pmatrix}, \quad F = \begin{pmatrix} 1 & & & & \\ -1 & 1 & 1 & & \\ -1 & & & 1 & 1 \end{pmatrix}, \quad e = f = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}.$$

Each sequence appears exactly once on the left hand side of the equations (2.3), accounting for the entry 1 in each column of E and F. The entry -1 in each row except the first stands for the right hand side in (2.3). Here, the polytope of realization plans has dimension two for each player.

For player 1, it is useful to consider only the nonnegative variables x_L, x_{RS}, x_{RT} which sum up to one since $x_\emptyset = 1$ and $x_R = x_{RS} + x_{RT}$. This defines a triangle as strategy space. The sequences L, RS, RT are here in one-to-one correspondence with the pure strategies $(L,*)$, (R,S), (R,T) in the RNF (in an obvious notation), so the simplex of mixed strategies is also two-dimensional.

When does the RNF lead to strategy spaces of higher dimension than the sequence form? This is the case when a player has *parallel* information sets h and h', that is, $\sigma_h = \sigma_{h'}$. Then, all combinations of moves at h and h' are part of separate strategies in the RNF. If there are no parallel information sets, there is a one-to-one correspondence between maximal sequences σ and RNF strategies, since no moves other than those in σ are relevant when playing σ. In that case (like for player 1 in the example), a player's mixed strategy simplex in the RNF has the same low dimension as his strategy space in the sequence form.

For player 2, there are two pairs y_a, y_b and y_c, y_d of strategic variables subject to $y \geq \mathbf{0}$ and $y_a + y_b = 1$ and $y_c + y_d = 1$. This strategy space is a square rather than a triangle. Its vertices correspond to the four pure strategies of player 2 (see also Figure 3.4 below). In the RNF, these correspond to the pure strategies (a,c), (a,d), (b,c), (b,d). Thus, the mixed strategy simplex of player 2 is a tetrahedron, of one dimension higher than the sequence form strategy space, because the two information sets of player 2 are parallel.

3. Optimal play

Sequence form payoffs are defined for pairs of sequences whenever these lead to a leaf, multiplied by the probabilities of chance moves on the path to the leaf. This defines two sparse matrices A and B of dimension $|S_1| \times |S_2|$ for player 1 and player 2, respectively. Then, the expected payoffs under the realization plans x and y are $x^\top Ay$ and $x^\top By$, representing the sum over all leaves of the payoffs at leaves multiplied by their realization probabilities.

We can characterize optimal play of a player by a pair of dual linear programs (LPs), as follows. If y is fixed, an optimal realization plan x of player 1 maximizes his expected payoff $x^\top(Ay)$, subject to $x \geq \mathbf{0}$, $Ex = e$. This LP has a dual LP with a vector u of unconstrained variables whose dimension is $1+|H_1|$, the number of rows of E. This dual LP is to minimize $e^\top u$ subject to

$$E^\top u \geq Ay. \tag{3.1}$$

For the extensive game in Figure 2.1, these constraints are indicated in Figure 3.1, where the rows and columns of $E^\top$ and A are marked with the components of x and u and y, respectively. The sparse payoff matrix A has blank (zero) entries for the pairs of sequences not leading to a leaf. Some zero entries are entered explicitly since they arise from payoffs that are zero. The payoff 24 at one of the leaves is entered as 12 in the matrix A since it is multiplied by the chance probability $1/2$ for reaching the leaf.

$$\begin{array}{c} \\ x_\emptyset \\ x_L \\ x_R \\ x_{RS} \\ x_{RT} \end{array}
\begin{array}{c} \begin{array}{ccc} u_0 & u_1 & u_2 \end{array} \\ \left[\begin{array}{ccc} 1 & -1 & \\ & 1 & \\ & 1 & -1 \\ & & 1 \\ & & 1 \end{array}\right] \end{array}
\quad \geq \quad
\begin{array}{c} \begin{array}{ccccc} y_\emptyset & y_a & y_b & y_c & y_d \end{array} \\ \left[\begin{array}{ccccc} & & & & \\ & 11 & 3 & & \\ & & & & \\ & 0 & 0 & 0 & 12 \\ & & & 6 & 0 \end{array}\right] \end{array}$$

Figure 3.1. Constraints (3.1) of best response LP for player 1 with dual vector u.

Primal and dual LP have the same optimal value of the objective function. That is, x is a best response to y iff there is a dual solution u with $x^\top(Ay) = e^\top u$. Since $e^\top = x^\top E^\top$, this is equivalent to

$$x^\top(E^\top u - Ay) = 0. \tag{3.2}$$

This is the *complementary slackness* of a pair of dual linear programs (see, for example, Schrijver, 1986).

Similarly, y is a best response to x iff $y \geq \mathbf{0}$, $Fy = f$, and there is an unconstrained dual vector v fulfilling

$$F^\top v \geq B^\top x \tag{3.3}$$

and

$$y^\top(F^\top v - B^\top x) = 0. \tag{3.4}$$

Figure 3.2 shows (3.3) in its transposed form $v^\top F \geq x^\top B$ for our example, so that like in Figure 3.1 rows and columns refer to sequences of player 1 and 2, respectively.

$$\begin{array}{c|ccccc|}
 & y_\emptyset & y_a & y_b & y_c & y_d \\
\hline
v_0 & 1 & & & & \\
v_1 & -1 & 1 & 1 & & \\
v_2 & -1 & & & 1 & 1 \\
\hline
\end{array}$$

$$\geq$$

$$\begin{array}{c|ccccc|}
\hline
x_\emptyset & & & & & \\
x_L & & 3 & 0 & & \\
x_R & & & & & \\
x_{RS} & & 0 & 5 & 2 & 0 \\
x_{RT} & & & & 0 & 1 \\
\hline
\end{array}$$

Figure 3.2. Constraints (3.3) of best response LP for player 2 with dual vector v.

The dual constraints (3.1) and (3.3) have the advantage that they stay linear even if the realization plan of the other player is treated as a variable, because the variables related to different players appear in different terms. These linear constraints and (2.4), together with the orthogonality conditions (3.2) and (3.4), define a *linear complementarity problem* (LCP) whose solutions (u,v,x,y) characterize the equilibria (x,y) of the game.

In our example, we can illustrate the solutions to this LCP by drawing the strategy spaces of the two players. Figure 3.3 shows the strategy space of player 1 consisting of the possible values of his strategic variables x_L, x_{RS}, x_{RT}. Figure 3.4 shows this for player 2 with the pairs y_a, y_b and y_c, y_d represented by the vertical and horizontal coordinates of a square. Note that $y_a = 0$ iff $y_b = 1$ and vice versa. The same holds for y_c and y_d. The redundant variables $x_\emptyset$, x_R and $y_\emptyset$ are not shown since their value is known, and they also have no payoff entry in Figure 3.1 and 3.2.

Figure 3.1 shows that the rows in (3.1) corresponding to the variables x_L, x_{RS}, x_{RT} have the form

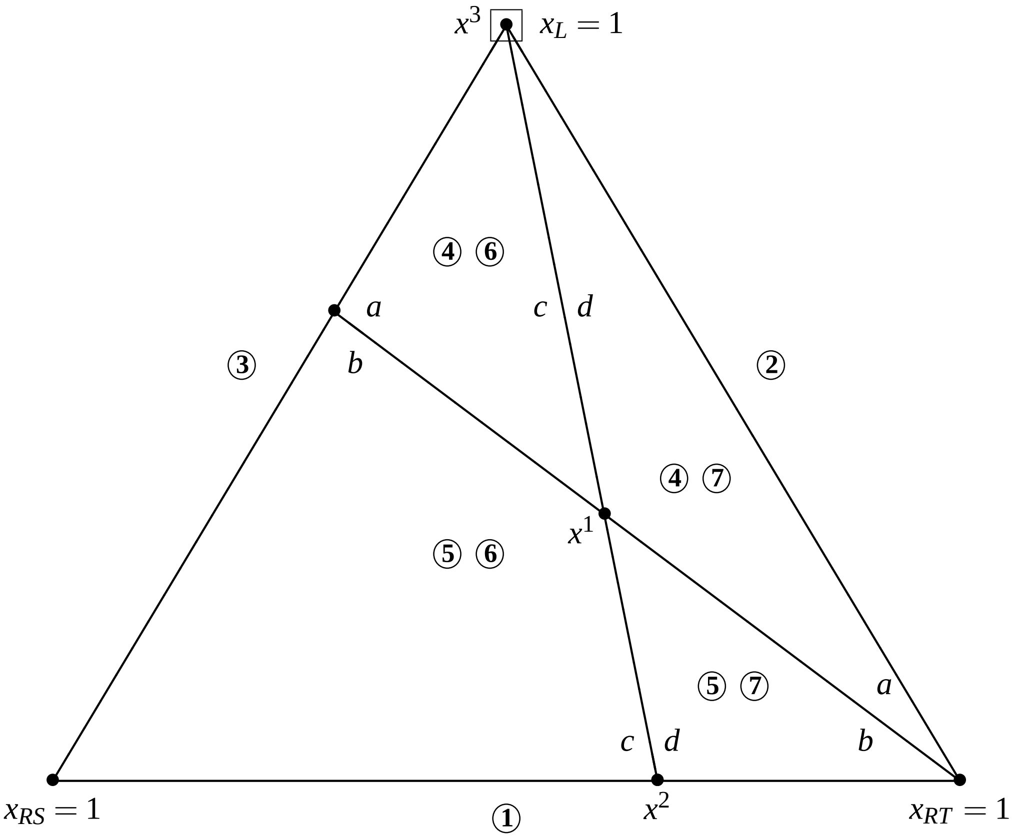

Figure 3.3. Strategy space of player 1 with best response sequences of player 2 for the sequence form of the game in Figure 2.1. The circled numbers are labels for identifying easily the equilibria (x^1, y^1), (x^2, y^2), (x^3, y^3).

$$
\begin{aligned}
u_1 &\geq 11y_a + 3y_b \\
u_2 &\geq 12y_d \\
u_2 &\geq 6y_c\,.
\end{aligned}
\tag{3.5}
$$

The two other rows read $u_0 - u_1 \geq 0$ and $u_1 - u_2 \geq 0$. Since $x_\emptyset = 1 > 0$, the corresponding inequality is always binding, that is, $u_0 - u_1 = 0$, by the slackness conditions (3.2). Similarly, $u_1 - u_2 = 0$ whenever $x_R > 0$, that is, when x_{RS} or x_{RT} is positive. Since x_L, x_{RS}, x_{RT} are not all zero, one inequality in (3.5) is binding and u_1 is the maximum of the right hand sides in (3.5). Furthermore, only for binding inequalities (where the maximum is achieved), which correspond to certain sequences σ of player 1, the component x_σ of x can be positive. In other words, only sequences σ that are "best responses" for player 1

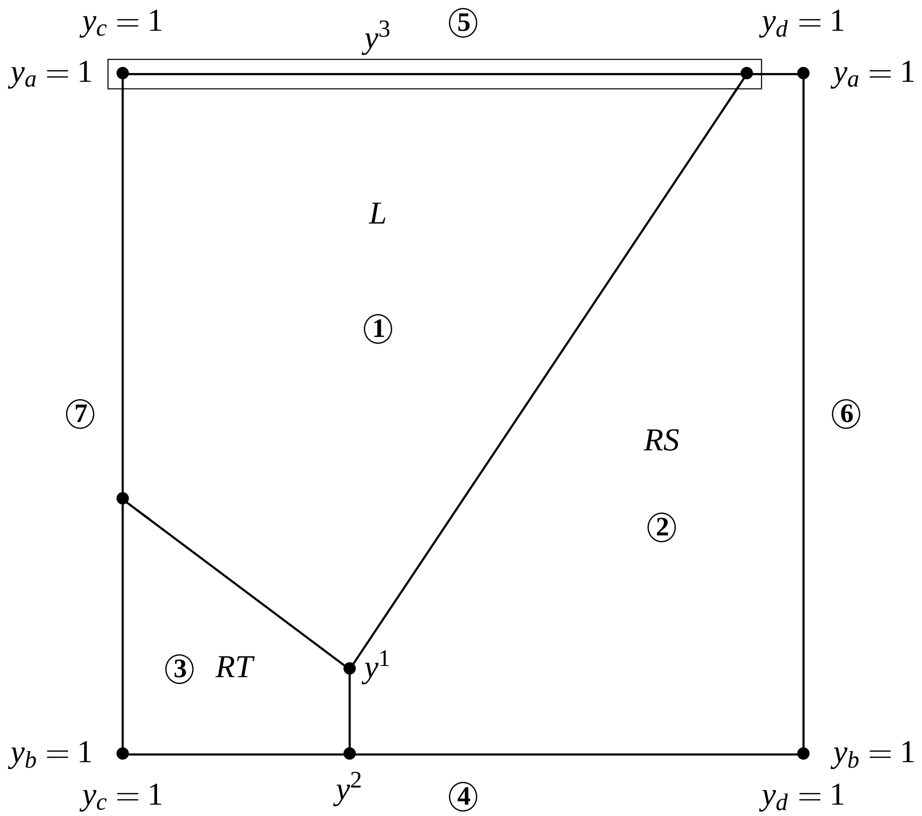

Figure 3.4. Strategy space of player 2 with best response sequences of player 1.

can have positive realization probability x_σ. For player 1, this is easy to interpret since σ corresponds to a pure strategy in the RNF. Figure 3.4 shows the regions where the sequences L (region ①), RS (region ②), or RT (region ③), are such best responses (the purpose of the circled numbers will be explained shortly). For example, (3.5) shows that RS is preferred to RT iff $12y_d \geq 6y_c = 6(1-y_d)$, that is, $y_d \geq 1/3$.

For player 2, interpreting the slackness conditions (3.4) is easiest if sequences are regarded as moves since player 2 does not pick a single sequence but one of each pair a,b and c,d (as expressed by the equations $Fy = f$ which are in this sense qualitatively different from $Ex = e$). As Figure 3.2 illustrates, $v_0 - v_1 - v_2 = 0$ since $y_\emptyset > 0$ and

$$
\begin{aligned}
v_1 \quad &\geq 3x_L \\
v_1 \quad &\geq \quad 5x_{RS} \\
v_2 &\geq \quad 2x_{RS} \\
v_2 &\geq \qquad x_{RT}
\end{aligned}
\tag{3.6}
$$

so that v_1 and v_2 have to be the maximum of the respective right hand sides in (3.6). Only when an inequality in (3.6) is binding, the respective move in each pair a, b and c, d can be played with positive probability. (The values of v_1 and v_2 represent partial payoffs at the information sets of player 2 where these moves take place, see von Stengel, 1996.)

Figure 3.3 shows the strategy space of player 1, subdivided into different regions where the sequences of player 2 are optimal. There are two pairs of regions, the regions ④ and ⑤ corresponding to the moves a and b, and the regions ⑥ and ⑦ corresponding to the moves c and d, where each point in the strategy space belongs to at least one region of each pair. The line separating the regions ④ and ⑤ describing where the move a or b is optimal depends by (3.6) only on the relative size of x_L versus x_{RS} and not on x_{RT}, so this line goes through the vertex $x_{RT} = 1$ of the triangle. This is due to the structure of the extensive game (the sequences a, b, RT occur in disjoint parts of the game tree) and is independent of the payoffs at the leaves. This represents a *degeneracy*, meaning that a move (here a or b) is optimal and played with zero probability at the same time (like for $x_{RT} = 1$, where the entire branch where a and b are played is omitted from play, so both moves are trivially optimal). For generic bimatrix games, such a degeneracy of the structure of best response regions can be excluded, here it is unavoidable. Similarly, the regions ⑥ and ⑦ where c and d are best responses are separated by a line through the vertex $x_L = 1$.

Just as the four strategies (a, c), (a, d), (b, c) and (b, d) of player 2 appear as vertices of his strategy space in Figure 3.4, they appear as intersections of best response regions in Figure 3.3 whenever both moves specified in the strategy are optimal. The strategy space of player 1 is thus divided into four such intersections, which would appear in the same way as best response regions when the strategies of player 2 were considered directly (so the degeneracies remain). The sequence form, however, gives here a more explanatory picture than the normal form of the game.

We can enumerate all equilibria of this game using a labeling technique similar to Shapley (1974). Each of the sequences L, RS, RT, a, b, c, d corresponds to a label ①, ..., ⑦ indicated by a circled number in Figure 3.3 and 3.4. There, these labels mark the (closed) regions inside the strategy spaces when they are best responses of the other player. Outside the strategy spaces, these labels mark the regions next to the facets of the own strategy space where the respective sequence has probability zero.

A strategy pair (x,y) in the two strategy spaces is labeled with all the labels of the regions that x and y belong to. For example, the pair of vertices $x_{RS} = 1$ and $y_b = 1$, $y_c = 1$ has the labels ①, ③, ⑤, ⑥, and ③, ④, ⑦. By complementary slackness, (x,y) is an *equilibrium* iff it has all seven labels since then every sequence is either a best response or played with probability zero. For the mentioned pair of vertices, this is not the case since the label ② (representing RS) is missing. One equilibrium, denoted (x^1,y^1), is in the interior of the two strategy spaces, $(x^1_L, x^1_{RS}, x^1_{RT}) = (5/14, 3/14, 3/7)$ and $(y^1_a, y^1_b, y^1_c, y^1_d) = (1/8, 7/8, 2/3, 1/3)$. Another equilibrium, denoted (x^2,y^2), is $(x^2_L, x^2_{RS}, x^2_{RT}) = (0, 1/3, 2/3)$ and $(y^2_a, y^2_b, y^2_c, y^2_d) = (0, 1, 2/3, 1/3)$. These equilibria are non-degenerate in the sense that no label occurs more than once. Finally, the vertex $x_L = 1$ in Figure 3.3 carries five labels ②, ③, ④, ⑥, ⑦, while the two missing labels ① and ⑤ mark two regions in Figure 3.4 which have a common boundary. All points on this boundary yield equilibria, indicated by the pair of rectangular boxes. These equilibria are given by $x^3_L = 1$, $y^3_a = 1$, and $0 \leq y^3_d \leq 11/12$. It is easy to check that there are no further equilibria.

Shapley (1974) explained the Lemke–Howson algorithm for bimatrix games with this labeling technique. Thereby, the edges separating the regions in Figure 3.3 and 3.4 define a graph. The outside regions are separated by edges which are connected to outer vertices $x = \mathbf{0}$ and $y = \mathbf{0}$ in an extra dimension. Starting from the ficticious equilibrium point $(x,y) = (\mathbf{0},\mathbf{0})$ which has all labels, the Lemke–Howson method follows a path in the (product) graph by dropping and picking up a label at a time until an equilibrium is reached. Such a method could be applied to our example, but with a more sophisticated change of labels since the graphs have non-uniform degree, partly due to degeneracies. Defining this method generally for the sequence form is a topic of future research. Instead, we will now describe a method of generating a path from an arbitrary starting point in each strategy space.

4. Complementary pivoting

The equilibrium conditions derived from the sequence form define a linear complementarity problem (LCP). The standard form of an LCP is characterized by an n-vector q and an $n \times n$ matrix M and requires to find n-vectors z and w so that

$$\begin{aligned} z &\geq \mathbf{0} \\ w = q + Mz &\geq \mathbf{0} \\ z^\top w &= 0\,. \end{aligned} \tag{4.1}$$

The condition $z^\top w = 0$ says that the nonnegative vectors $z = (z_1, \ldots, z_n)^\top$ and $w = (w_1, \ldots, w_n)^\top$ are *complementary*, that is, at least one variable of each pair z_i, w_i for $1 \leq i \leq n$ is zero, whereas the other may be nonzero. A *mixed* LCP (see Cottle, Pang, and Stone, 1992, p. 29) has more general constraints than $z, w \geq \mathbf{0}$ in (4.1): some variables z_i may be unrestricted in sign where the corresponding variable w_i is always zero. The LCP derived from the sequence form with constraints (2.4) and (3.1)–(3.4) is such a mixed LCP. It has variables $z = (u, v, x, y)^\top$ and $w = (w_u, w_v, w_x, w_y)^\top$. The sign restrictions are none for u and v, whereas $x \geq \mathbf{0}$, $y \geq \mathbf{0}$, $w_u = \mathbf{0}$, $w_v = \mathbf{0}$, $w_x \geq \mathbf{0}$, $w_y \geq \mathbf{0}$. The requirement $z^\top w = 0$ is then equivalent to

$$x^\top w_x = 0, \qquad y^\top w_y = 0. \tag{4.2}$$

The dimension of the LCP is $n = 2 + |H_1| + |H_2| + |S_1| + |S_2|$. The LCP data are

$$q = \begin{pmatrix} -e \\ -f \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \qquad M = \begin{pmatrix} & & E & \\ & & & F \\ E^\top & & & -A \\ & F^\top & -B^\top & \end{pmatrix}.$$

It is easy, but computationally not necessary, to convert this mixed LCP to the standard form (4.1) by representing u and v as differences of nonnegative vectors and replacing each equation in $Ex = e$ and $Fy = f$ by a pair of inequalities (see, for example, Koller, Megiddo, and von Stengel, 1996).

Lemke (1965) described an algorithm for solving the LCP (4.1). It uses an additional n-vector d, called *covering vector*, and a corresponding scalar variable z_0, and computes with solutions to the augmented system

$$\begin{aligned} z\,, \quad z_0 &\geq \mathbf{0} \\ w = q + Mz + dz_0 &\geq \mathbf{0} \\ z^\top w &= 0\,. \end{aligned} \tag{4.3}$$

An *almost complementary basis* is a set of n *basic* variables that contains at most one variable of each complementary pair z_i, w_i for $1 \leq i \leq n$ and possibly z_0 such that these variables define a unique solution to $w = q + Mz + dz_0$ if all other (*nonbasic*) variables are zero. Suppose this solution fulfills (4.3), that is, $z \geq \mathbf{0}$, $z_0 \geq 0$, and $w \geq \mathbf{0}$. If z_0 is nonbasic, this solves (4.1). Otherwise, there is a pair z_i, w_i of nonbasic variables. Allowing one of them, designated as *entering variable*, to be nonnegative besides the basic variables, the solutions of (4.3) to these $n+1$ variables usually define a line segment that joins the current almost complementary basis to another one. That new almost complementary basis contains the entering variable, and some formerly basic variable that has become zero has *left* the basis. If this is not z_0, its complement is the next entering variable. The resulting iterative change of bases is called *complementary pivoting*. After a suitable initialization, this generates a sequence of bases that define a piecewise linear path which, under certain conditions, ends in a solution to the LCP (4.1). Koller, Megiddo, and von Stengel (1996) give a more detailed exposition of Lemke's algorithm and show that it terminates for the LCP derived from the sequence form.

For the specific problem at hand, we choose a particular covering vector d that is related to the starting position for our computation. Let (s,t) be an arbitrary *starting vector*, that is, a pair of realization plans for the two players, so that

$$s \geq \mathbf{0}, \quad Es = e, \qquad t \geq \mathbf{0}, \quad Ft = f, \tag{4.4}$$

and let

$$d = \begin{pmatrix} e \\ f \\ -At \\ -B^\top s \end{pmatrix}. \tag{4.5}$$

The sign constraints for our mixed LCP and the equations $w = q + Mz + dz_0$ have then the form

$$
\begin{array}{rcccl}
 & x, & y, & & z_0 \geq \mathbf{0}, \\
 & Ex & + & & ez_0 = e \\
 & & Fy+ & & fz_0 = f \\
w_x = E^\top u & & -Ay- & & (At)z_0 \geq \mathbf{0} \\
w_y = & F^\top v - B^\top x & & & -(B^\top s)z_0 \geq \mathbf{0}.
\end{array}
\tag{4.6}
$$

An initial solution to (4.6), which fulfills also (4.2), is given by $z_0 = 1$, $x = \mathbf{0}$, $y = \mathbf{0}$, and suitable vectors u and v so that $E^\top u \geq At$ and $F^\top v \geq B^\top s$, that is, $w_x \geq \mathbf{0}$ and $w_y \geq \mathbf{0}$. We also have to find an almost complementary basis representing this initial solution. There, the problem is that some components of x and y have to be taken as basic variables, although with value zero, since otherwise the linear system (4.6) restricted to the basic variables does not have full rank n. We will address this question shortly.

The conditions (4.6) and (4.2) are the equivalent of (4.3) and hold for all points on the piecewise linear path computed by Lemke's algorithm. The following lemma and the discussion thereafter shows that this path induces a path in the product of the two strategy spaces which begins at the starting vector (s,t) and ends at an equilibrium.

Lemma 4.1. *In any solution (u,v,x,y,z_0) to (4.6), $x + sz_0$ is a realization plan for player 1, $y + tz_0$ is a realization plan for player 2, and $x_\emptyset = y_\emptyset = 1 - z_0 \geq 0$.*

Proof. The constraints (4.6) and (4.4) imply $x + sz_0 \geq \mathbf{0}$, $y + tz_0 \geq \mathbf{0}$,

$$E(x + sz_0) = Ex + (Es)z_0 = Ex + ez_0 = e,$$

and similarly $F(y + tz_0) = f$. By (2.3), the first of each of these equations says $x_\emptyset + z_0 = 1$ and $y_\emptyset + z_0 = 1$, respectively. □

By Lemma 4.1, any solution to (4.6) fulfills $0 \leq z_0 \leq 1$. We can regard z_0 as a probability assigned to the starting vector, initially $z_0 = 1$. The algorithm terminates when $z_0 = 0$, so that x and y are realization plans and (x,y) is an equilibrium by (4.2). At intermittent steps of the computation with $0 < z_0 < 1$, the pair $(x + sz_0, y + tz_0)$ of realization plans can be seen as a mixture of a pair $(\bar{x},\bar{y})$ and the starting pair (s,t), chosen with probabilities $1 - z_0$ and z_0, respectively. Namely, let $\bar{x} = x \cdot 1/(1 - z_0)$ and

$\bar{y} = y \cdot 1/(1-z_0)$, so that

$$x + sz_0 = \bar{x}(1-z_0) + sz_0, \qquad y + tz_0 = \bar{y}(1-z_0) + tz_0. \tag{4.7}$$

By (4.6), $Ex = e(1-z_0)$ and $Fy = f(1-z_0)$, which implies that $\bar{x}$ and $\bar{y}$ are realization plans.

The realization plan $x + sz_0$ of player 1 plays any sequence σ in S_1 with at least z_0 times the probability s_σ it has under s, since $x \geq \mathbf{0}$. Whenever $x_\sigma > 0$, the sequence σ has a larger probability than the probability $s_\sigma z_0$. Similarly, the realization plan $y + tz_0$ selects some sequences σ in S_2 with probability $t_\sigma z_0$ if $y_\sigma = 0$, the others with larger probability $y_\sigma + t_\sigma z_0$. The positive components x_σ and y_σ of x and y are the same as the positive components of $\bar{x}$ and $\bar{y}$ in (4.7), up to scalar multiplication with $1 - z_0$. By the following lemma, these are best response sequences to the current pair of realization plans.

Lemma 4.2. *Consider a solution (u, v, x, y, z_0) to (4.6) and (4.2) with $z_0 < 1$, and let $\bar{x} = x \cdot 1/(1-z_0)$ and $\bar{y} = y \cdot 1/(1-z_0)$. Then $(\bar{x}, \bar{y})$ is a pair of realization plans where $\bar{x}$ is a best response to $y + tz_0$ and $\bar{y}$ is a best response to $x + sz_0$.*

Proof. As shown above, (4.6) implies that $\bar{x}$ and $\bar{y}$ are realization plans, and

$$E^\top u \geq A(y + tz_0).$$

By (4.2), $\bar{x}^\top (E^\top u - A(y + tz_0)) = \bar{x}^\top w_x = 0$, which is the complementary slackness condition (3.2) with $y + tz_0$ instead of y showing that $\bar{x}$ is a best response to $y + tz_0$ (and u a corresponding optimal dual solution). Similarly, $\bar{y}$ is a best response to $x + sz_0$. $\square$

In order to leave the starting vector (s, t), we need solutions to (4.6) and (4.2) where $z_0 < 1$ is possible. Whenever z_0 decreases from 1, the conditions (2.3) for realization plans imply that usually *several* components of x (and similarly y) become simultaneously nonzero in the equations $Ex = e(1-z_0)$, since these are the same homogeneous equations as $Ex = e$ in (2.3), and only the first, nonhomogeneous equation $x_\emptyset = 1 - z_0$ is different. The initial solution $x = \mathbf{0}$, $y = \mathbf{0}$ does not show which components of x and y should be increased. One of these components is the first entering variable, the others must belong to the initial almost complementary basis. We determine this basis by linear programming, similarly to Kamiya and Talman (1990) and Dai and Talman (1993), as follows.

Our initialization step is motivated by Lemma 4.2. Compute a best response $\bar{x}$ to t and a best response $\bar{y}$ to s. That is, $\bar{x}$ is a solution to the LP: maximize $x^\top(At)$ subject to $Ex = e$, $x \geq \mathbf{0}$, and $\bar{y}$ to the LP: maximize $(s^\top B)y$ subject to $Fy = f$, $y \geq \mathbf{0}$. This yields also corresponding optimal dual vectors u and v so that $\bar{x}^\top(E^\top u - At) = 0$ and $\bar{y}^\top(F^\top v - B^\top s) = 0$. We can assume that $\bar{x}$ and $\bar{y}$ are *basic solutions* to these two LPs, for example as they are computed by the simplex algorithm for linear programming. That is, an invertible submatrix of each matrix E and F (which both have full row rank) determines the respective basic components $\bar{x}_\sigma$ and $\bar{y}_\sigma$ which may be positive, and determines uniquely u and v, respectively. Then, the initial almost complementary basis for Lemke's algorithm contains z_0, all components of u and v, all but one of the variables x_σ and y_σ corresponding to the basic LP variables $\bar{x}_\sigma$ and $\bar{y}_\sigma$ above (the missing one is the first entering variable), and the slack variables $(w_x)_\sigma$ and $(w_y)_\sigma$ in $w_x = E^\top u - At$ and $w_y = F^\top v - B^\top s$ for the other sequences σ. Later, we come back to some fine points concerning this initialization step. We summarize what we have found so far.

Algorithm 4.3. *Consider an extensive game for two players with perfect recall, and its sequence form with payoff matrices A and B and constraint matrices E and F for player 1 and player 2, respectively. Choose a starting vector (s,t) fulfilling (4.4). Construct the mixed LCP with constraints (4.6) and (4.2). Solve this LCP as follows.*

(a) *Find an initial almost complementary basic solution with $z_0 = 1$ where the basic variables are z_0, all components of u and v, and all but one of the components of x and y representing best response sequences against t and s, respectively.*

(b) *Iterate by complementary pivoting steps applied to pairs $x_\sigma, (w_x)_\sigma$ or $y_\sigma, (w_y)_\sigma$ of complementary variables.*

(c) *As soon as z_0 becomes zero, let z_0 leave the basis and pivot. Terminate. The computed equilibrium is (x,y).*

We have shown in Lemma 4.1 that in the course of the computation, the values of x, y, and z_0 determine always a pair of realization plans $x + sz_0$ and $y + tz_0$ and thus a path in the product of the two strategy spaces. We are only interested in this path, since the basic variables in u and v are uniquely determined.

It remains to show that the algorithm *terminates*. With the above interpretation, we can exclude *ray termination*, which may cause Lemke's algorithm to fail, because the path cannot leave the strategy space. Before, this was proved by Koller, Megiddo, and von Stengel (1996) using a rather technical theorem. Thus, the algorithm terminates if the path is unique in the sense that no basis is revisited. This requires systematic *degeneracy resolution*. We discuss this technical topic in the Appendix.

5. Finding equilibria

Our complementary pivoting algorithm originates with the algorithm by van den Elzen and Talman (1991) for bimatrix (i.e., generic two-person normal form) games. They consider a starting vector in the product of the two mixed strategy simplices which are the players' strategy spaces. A bimatrix game can be represented by the sequence form if A and B are the payoff matrices and both E and F are just rows of ones and $e = f = 1$, so that (2.4) says that x and y are mixed strategies. With the covering vector d in (4.5), one can see with the help of Lemma 4.1 that Lemke's algorithm above is in fact equivalent to the algorithm by van den Elzen and Talman. For a general sequence form, the shape of the strategy spaces is new. We will illustrate this aspect with our example.

The starting vector (s,t) is used throughout the computation for reference, since it determines the system (4.6). As mentioned, the first step is to find a pair $(\bar{x},\bar{y})$ of best responses to (s,t). Like van den Elzen and Talman (1991), we assume that these best responses are *unique*, so that every optimal move (in a sequence σ with positive probability $\bar{x}_\sigma$ or $\bar{y}_\sigma$) is the only optimal one at its information set. This assumption (which can be relaxed, see the Appendix) is true for a generic starting vector. Thus, $\bar{x}$ and $\bar{y}$ represent pure strategies, which are vertices of the strategy spaces of player 1 and 2, respectively.

Consider the line segment that joins (s,t) to $(\bar{x},\bar{y})$. This is the set of pairs $(\bar{x}(1-z_0)+sz_0, \bar{y}(1-z_0)+tz_0)$ for $0 \le z_0 \le 1$. An initial part of this line segment, where (x,y) corresponds to $(\bar{x},\bar{y})$ by (4.7) and z_0 assumes values in some interval $[\bar{z}_0, 1]$, is the first piece of the path computed by the algorithm. The $n+1$ variables (initial basic variables and entering variable) whose solutions to (4.6) determine this first line segment include z_0, all variables x_σ and y_σ for the sequences σ that are best responses to t and s, respectively, and the slack variables $(w_x)_\sigma$ and $(w_y)_\sigma$ in $w_x = E^\top u - A(y+tz_0)$ and $w_y = F^\top v - B^\top(x+sz_0)$ for some of the nonoptimal sequences σ. Because the best response sequences are unique, all these slacks are *positive* for $z_0 = 1$ and therefore stay positive if z_0 is slightly smaller than one.

As a side remark concerning the initialization step 4.3(a), note that we allow some basic variables x_σ or y_σ for nonoptimal sequences σ (which have value zero and do not matter) in order to get the necessary number of basic variables. This happens if certain information sets are irrelevant for the best responses $\bar{x}$ and $\bar{y}$. Technically speaking, the strategy spaces may not be simple polytopes; as one consequence, we cannot use the algorithm by Dai and Talman (1993) for our problem.

We now illustrate the progress of the algorithm with our example above. We choose the following starting vector (s,t), omitting as before the components $s_\emptyset$, s_R, and $t_\emptyset$:

$$(s_L, s_{RS}, s_{RT}) = (3/10, 7/20, 7/20) \qquad \text{and} \qquad (t_a, t_b, t_c, t_d) = (1/3, 2/3, 1/3, 2/3).$$

The starting points s and t are marked in Figure 5.1 and 5.2 by a dot in the interior of each strategy space. The unique best response sequence of player 1 to t is RS, and the unique best response sequences of player 2 to s are b and c, which defines the vertices $\bar{x}$ and $\bar{y}$, denoted by RS and b,c in Figure 5.1 and 5.2, respectively. Thus, two variables of x_{RS}, y_b, y_c are basic and one of them is the first entering variable. The other components of x and y are nonbasic. The algorithm makes the following steps, indicated in the figures.

1. The first step is the line segment starting at (s,t) in direction $(\bar{x},\bar{y})$, where with (4.7) the two realization plans $x+sz_0$ and $y+tz_0$ of the two players depend jointly on z_0. That is, the two arrows marked “1.” are traversed simultaneously by reducing z_0 from 1 to $\bar{z}_0 = 9/16$, where the path hits the best response region for the sequence L of player 1 in Figure 5.2. In terms of the system (4.6), this means that the slack $(w_x)_L$ of the payoff for that sequence becomes zero.

In Figure 5.2, the end of the arrow “1.” is the corner of a square, a smaller-sized replica of the strategy space containing the starting point t at the same relative position. In other

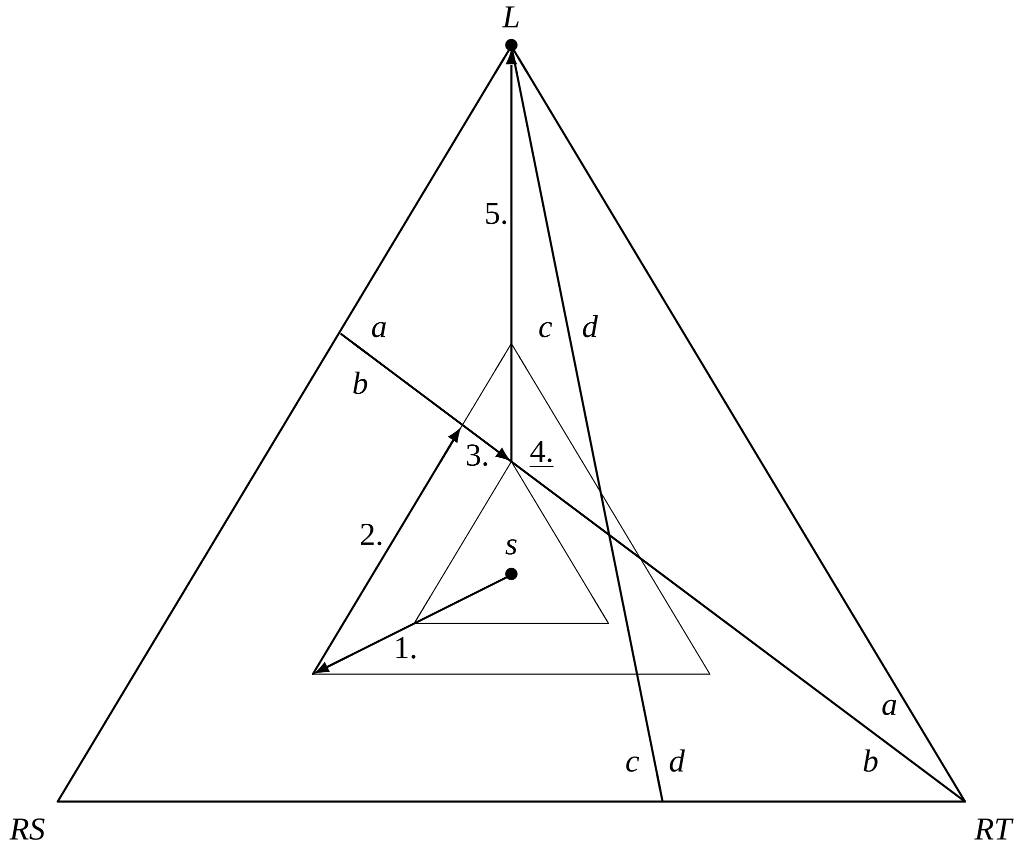

Figure 5.1. Strategy space of player 1 with best response sequences of player 2 and computation steps, indicated by arrows or (underlined) steps with no change for player 1. The starting point s is $(s_L, s_{RS}, s_{RT}) = (3/10, 7/20, 7/20)$.

words, we expand a set from t towards the corners of the strategy space of player 2. This set contains all realization plans of player 2 where each sequence σ is played at least with probability $t_\sigma \bar{z}_0$. At that corner, only the sequences b and c of player 2 have positive components y_b and y_c, whereas $y_a = y_d = 0$. Similarly, the end of the arrow "1." in Figure 5.1 is the corner of a triangle which contains all realization plans of player 1 where each sequence σ is played at least with probability $s_\sigma \bar{z}_0$. There, only x_{RS} is positive, the other sequences are played with minimum probability.

2. Since the slack $(w_x)_L$ has become zero, it is replaced by its complementary variable x_L that is now increased. This is the complementary pivoting step (b) of Algorithm 4.3 where $(w_x)_L$ leaves and x_L enters the basis. When x_L is increased, then z_0 cannot decrease further, since this would make RS nonoptimal, or increase, since

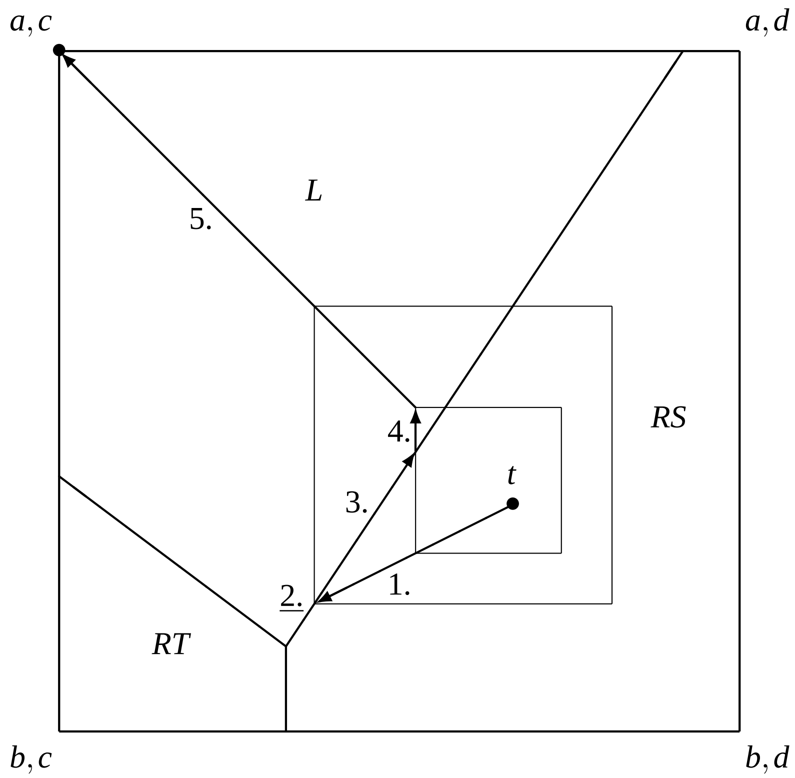

Figure 5.2. Strategy space of player 2 with best responses of player 1 and computation steps. The starting point t is $(t_a, t_b, t_c, t_d) = (^1\!/_3, ^2\!/_3, ^1\!/_3, ^2\!/_3)$.

this would make L nonoptimal (see Figure 5.2), but both x_{RS} and x_L are basic and may have positive values. So z_0 stays as it is. Furthermore, only y_b and y_c are basic for player 2, so his position in the corner of the square is unchanged, marked with "2." (underlined) in Figure 5.2. For player 1, the arrow "2." in Figure 5.1 denotes a relative increase of x_L until the best response set of the sequence a of player 2 is reached. Then, the basic slack variable $(w_y)_a$ becomes zero and is exchanged with y_a.

3. Currently, the variables x_L and x_{RS} of player 1 are basic, so that $(w_x)_L$ and $(w_x)_{RS}$ are nonbasic and zero. For player 2, the computed path in Figure 5.2 is therefore now a segment of the common boundary of the two best response regions for L and RS. Thereby, the relative size of y_a can only increase if z_0 is *increased*, which *shrinks* the set of realizations plans where each sequence σ has minimum probability $t_\sigma z_0$. This generates a smaller square in Figure 5.2, and, by the same shrinking factor, a smaller triangle in Figure 5.1, until x_{RS} becomes zero, which happens when z_0 is

increased to $^{60}/_{77}$. That is, the end of the arrow “3.” points to the tip of the small triangle where x_L is the only positive component of x. The leaving variable x_{RS} is replaced by its complement $(w_x)_{RS}$ which enters the basis, so that the best response region for RS in Figure 5.2 can be left.

4. Nothing changes for player 1 in Figure 5.1 since y_a, y_b, y_c are all basic, so z_0 remains constant. Leaving the best response region for RS means that y_a is increased, until y_b is zero. This variable is replaced by its complement $(w_y)_b$. The best response region for b in Figure 5.1 can be left.

5. The current basis contains only x_L, y_a, y_c, so the best response sequences are L for player 1 and a and c for player 2. Then, z_0 can be decreased again, in fact until $z_0 = 0$, reaching the equilibrium (x^3, y^3) with $x_L^3 = 1$ and $y_a^3 = y_c^3 = 1$, which is the end of the computed path.

This example is specifically designed to show that an intermittent increase of z_0 is possible, which is usually rare, at least for low-dimensional strategy spaces or for bimatrix games. This behavior, and which equilibrium is found in case the game has more than one equilibrium, depend on the starting vector. The reader may verify that if the starting point t of player 2 is changed so that all moves a, b, c, d have equal probability $^1/_2$, then the same equilibrium is reached in three steps, first moving in direction L in Figure 5.1 and direction b, c in Figure 5.2, until a becomes a best response of player 2, then shifting from b to a while z_0 is fixed, and finally reaching the equilibrium. For yet another starting vector where, in addition, s is changed so that all moves of player 1 have equal probability, that is, $s_L = s_R = {}^1/_2$ and $s_{RS} = s_{RT} = {}^1/_4$, the same equilibrium is reached in a single step.

By changing the starting vector it is also possible to compute other equilibria. For example, when the algorithm starts from $(\bar{s}, \bar{t})$ with $(\bar{s}_L, \bar{s}_{RS}, \bar{s}_{RT}) = ({}^3/_{10}, {}^7/_{20}, {}^7/_{20})$ and $(\bar{t}_a, \bar{t}_b, \bar{t}_c, \bar{t}_d) = ({}^1/_8, {}^7/_8, {}^1/_3, {}^2/_3)$, then it computes the equilibrium (x^2, y^2) (see Figures 3.3 and 3.4).

All the equilibria reachable in this manner have a negative *index* (the index of an equilibrium is the sign of the determinant of a certain matrix related to that equilibrium; see Shapley, 1974; van der Laan, 1984). However, it is also possible to find positively indexed equilibria. Crucial here is the observation that the set of realization plans satisfying (4.6) and (4.2) is in general larger than the piecewise linear path connecting the starting vector and the related equilibrium. This set includes all other Nash equilibria. In the first example above with starting vector (s,t), there exists also a piecewise linear path of vectors connecting the equilibria (x^1,y^1) and (x^2,y^2). All these strategy vectors obey the conditions demanded for by the algorithm. We can therefore find the positively indexed equilibrium (x^1,y^1) as follows. In the first stage we find the equilibrium (x^2,y^2) when starting from $(\bar{s},\bar{t})$. Then we consider the system (4.6), (4.2) derived from the first starting vector (s,t). The equilibrium (x^2,y^2) is a solution to this system which is not on the path starting at (s,t). Starting with $(x,y)=(x^2,y^2)$ and $z_0=0$, we start the algorithm by letting z_0 enter the basis. Then the algorithm computes the equilibrium (x^1,y^1) by the following steps.

1′. Basic variables are $(w_x)_L$, x_{RS}, x_{RT}, $(w_y)_a$, y_b, y_c, y_d. The entering variable z_0 is increased to $^3/_8$, where $(w_x)_L$ becomes zero and leaves the basis.

2′. The entering variable x_L is increased to $^{137}/_{560}$, where $(w_y)_a$ becomes zero and leaves. No change occurs for z_0 and y.

3′. The entering variable y_a is increased until $y_a = {}^1/_8$ where $z_0=0$. Then z_0 leaves the basis. The algorithm terminates with $(x,y)=(x^1,y^1)$.

In principle, the equilibrium (x^1,y^1) is computed at the end of step 2′ where $x^1 = x+sz_0$ and $y^1=y+tz_0$. Since the path ends, van den Elzen and Talman (1991, 1995) let the algorithm terminate here. We run the method as a special case of Lemke's algorithm and include the final step 3′. Driving z_0 to zero is also appropriate for the equivalence with Harsanyi and Selten's tracing procedure, where z_0 is the probability of playing against the prior (s,t). This is the topic of the next section.

6. Game-theoretic interpretation

The computation of the algorithm can be interpreted game-theoretically. We will show that it mimicks the *tracing procedure* by Harsanyi and Selten (1988), applied to the normal form of the game. Furthermore, if the starting vector is completely mixed, then the computed equilibrium will be *normal form perfect*. For greater detail we refer to van den Elzen and Talman (1995) who showed these properties for bimatrix games.

The tracing procedure starts from a common *prior*. For two-person games, this is a pair of strategies describing the preconceptions of the players about the other player's behavior. Initially, the players react optimally to these expectations. In general they observe that their expectations are not fulfilled and thus adjust their expectations about the behavior of the other player. Besides, more and more information about the game is revealed. By simultaneously and gradually adjusting the expectations and reacting optimally against these revised expectations, eventually an equilibrium is reached.

Consider first a bimatrix game Γ with payoff matrices A and B, as a special case of a game in sequence form. The tracing procedure generates a path of strategy pairs $(\bar{x},\bar{y})$. Each such pair is an equilibrium in a parameterized game $\Gamma(z_0)$. The prior is the same as our starting vector (s,t). The payoffs in $\Gamma(z_0)$ are as if each player plays with probability z_0 against the prior and with probability $1-z_0$ against the actual strategy in $(\bar{x},\bar{y})$ of the other player. That is, player 1 receives in $\Gamma(z_0)$ expected payoff $z_0 \cdot \bar{x}^\top(At) + (1-z_0) \cdot \bar{x}^\top(A\bar{y})$, and player 2 receives payoff $z_0 \cdot (s^\top B)\bar{y} + (1-z_0) \cdot (\bar{x}^\top B)\bar{y}$.

The tracing procedure starts with $z_0 = 1$, where $\bar{x}$ and $\bar{y}$ are the players' optimal responses to the prior. Then z_0 is decreased, changing $(\bar{x},\bar{y})$ such that it stays an equilibrium of $\Gamma(z_0)$. Sometimes, changing the value of z_0 may stall (similar to the computation steps 2. and 4. in the example in Section 5), tracing then instead a continuum of equilibria in $\Gamma(z_0)$ which is usually one-dimensional (the non-standard case is discussed in Schanuel, Simon, and Zame, 1991). These conditions define the so-called *linear tracing procedure*. They generate a unique path except for degeneracies, which, given any starting vector, do not occur for a bimatrix game with generic payoffs. Degeneracies are resolved by the *logarithmic tracing procedure* (see Harsanyi and Selten, 1988), which we do not regard here. The procedure ends with $z_0 = 0$ where $\Gamma(0) = \Gamma$.

Generically, the linear tracing procedure suffices and coincides with the logarithmic tracing procedure. Then, it corresponds to the computation by Algorithm 4.3. In $\Gamma(z_0)$, player 1 receives payoffs that are the same as the original payoffs in Γ against the randomized strategy $z_0 \cdot t + (1 - z_0) \cdot \bar{y}$. This holds because we are dealing with only two players. Similarly, player 2 receives the original payoffs against the strategy $z_0 \cdot s + (1 - z_0) \cdot \bar{x}$. By (4.7), this is the strategy pair $(x + sz_0, y + tz_0)$ where the pair (x, y) is computed by the algorithm and corresponds to $(\bar{x}, \bar{y})$. Lemma 4.2 asserts that $(\bar{x}, \bar{y})$ is an equilibrium of $\Gamma(z_0)$.

Thus, the paths generated by the tracing procedure and by the algorithm coincide up to projection. Whereas the tracing procedure traces a pair of strategies in the full strategy space and considers convex combinations of *payoffs* with weights z_0 and $1 - z_0$, the complementary pivoting algorithm generates the corresponding convex combinations of *strategies* which belong to a restricted strategy set that expands and shrinks proportionally to $1 - z_0$. In other words, the tracing procedure generates a path of Nash equilibria for games with perturbed payoffs, whereas the algorithm delivers a path of Nash equilibria related to restricted strategy domains. Both methods terminate in the same equilibrium.

The preceding statements apply directly to bimatrix games, as shown by van den Elzen and Talman (1995). By the same arguments, the computation emulates a suitably defined tracing procedure for the sequence form. Moreover, we can show that the piecewise linear path computed by the algorithm applies also to the *normal form* of the game, where the strategy space of each player is the simplex of his mixed strategies. So far, the computed path lies in the sequence form strategy space. In Section 2 we have shown that this strategy space is the image of the simplex of mixed strategies under the linear map defined by (2.2). A suitable pre-image under this map of the computed path in the sequence form strategy space yields a piecewise linear path in mixed strategies, as follows.

Consider the *endpoints* of each line segment of the computed path (defined by two successively computed almost complementary bases, see Section 4). Denote the endpoints of such a line segment by $(\tilde{x}, \tilde{y})$ and $(\hat{x}, \hat{y})$, say. Let S be the line segment connecting $\tilde{x}$ and $\hat{x}$ in the strategy space of player 1 (the consideration for player 2 is similar). Consider mixed strategies $\tilde{\mu}$ and $\hat{\mu}$ of player 1 that have realization plans $\tilde{x}$ and $\hat{x}$, respectively. In the mixed strategy simplex of player 1, the line segment connecting $\tilde{\mu}$ and $\hat{\mu}$ is mapped under (2.2) to S since that map is linear. Thus, S can indeed be translated to a line segment in mixed strategies.

The particular pre-image of S in the mixed strategy simplex does not matter, because mixed strategies with the same realization plans are realization equivalent and therefore payoff equivalent, so the equilibrium property in $\Gamma(z_0)$ is preserved. A canonical choice for $\tilde{\mu}$ and $\hat{\mu}$ are the behavior strategies of player 1 with realization plans $\tilde{x}$ and $\hat{x}$, respectively. Note, however, that the entire line segment S should not be translated to behavior strategies, since this does not yield a line in the mixed strategy simplex if the convex combinations of $\tilde{\mu}$ and $\hat{\mu}$ are not behavior strategies.

We have shown that we can consider the computed path as a trace in mixed strategies. Following van den Elzen and Talman (1995), we can also show that the computed equilibrium is normal form perfect if the prior (s,t) is *completely mixed*. A completely mixed realization plan assigns positive realization probability to every sequence. The corresponding behavior strategy plays every move with positive probability, and considered as a mixed strategy, it chooses every pure strategy with positive probability.

Lemma 6.1. *If the starting vector (s,t) is completely mixed, then Algorithm 4.3 computes an equilibrium that is normal form perfect.*

Proof. Let the starting vector (s,t) be completely mixed and let (x^*,y^*) be the computed equilibrium. Except for its endpoint (x^*,y^*), the last line segment of the computed path consists of pairs $(x+sz_0, y+tz_0)$ of realization plans where $z_0 > 0$, due to condition 4.3(c). Therefore, these realization plans are, like s and t, completely mixed. The equilibrium (x^*,y^*) is the *limit* of these realization plans when z_0 goes to zero, and is a pair of *best responses* to these realization plans because of the complementarity condition (4.2), since x^* and y^* have the same basic (that is, positive) components as x and y (a similar argument was made in the proof of Lemma 4.2). These properties hold also when the computed path is translated to mixed strategies as described above. According to Selten (1975, Thm. 7), they imply that the equilibrium (x^*,y^*) is perfect in the normal form. $\square$

For bimatrix games, each point on the computed path translates to an equilibrium of the *restricted* game where each strategy is played at least with the probability it has under $(s,t)\cdot z_0$. This can also serve to prove that the equilibrium is perfect. Using the sequence form, it is better to invoke Selten's condition as in the preceding proof since the probabilities for playing pure strategies may vary for realization equivalent mixed strategies and are therefore not well defined.

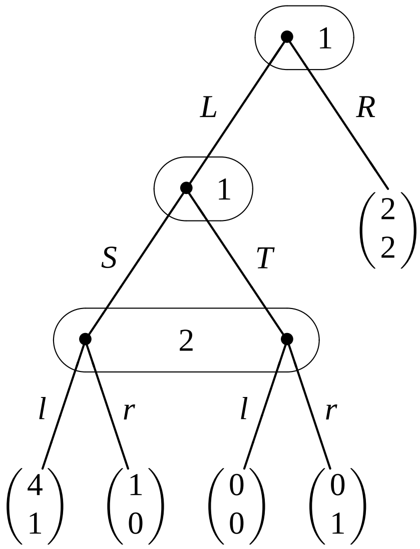

Figure 6.1. Extensive game where the equilibrium (R, r) is normal form perfect but not extensive form perfect.

The tracing procedure, and the concept of a perfect equilibrium, are different when they are defined for the extensive form instead of the normal form of the game. The extensive game in Figure 6.1, taken from van Damme (1987, p. 114), has an equilibrium (R, r) in pure strategies in the reduced normal form. In the full normal form, the equilibrium would be written (RS, r). This equilibrium is not extensive form perfect since it is not even subgame perfect. The only subgame perfect equilibrium is (LS, l). However, the equilibrium (R, r) is normal form perfect. It is computed by Algorithm 4.3 when started from the prior (s, t) where $(s_{LS}, s_{LT}, s_R) = (1/4, 1/2, 1/4)$ and $(t_l, t_r) = (1/5, 4/5)$, say. The dominated sequence (and strategy) LT has probability zero in the equilibrium (R, r). However, the sequence LS has also probability zero, and in approaching the equilibrium, the probabilities for the *moves* S and T are as prescribed in the starting vector, which implies a non-vanishing probability for the dominated move T. Only at the equilibrium these move probabilities become undefined.

"Unreasonable" behavior at unreached information sets cannot be excluded with normal form approaches. Computing with the sequence form is such an approach. However, it is trivial to solve subgames first. In Figure 6.1, this removes the equilibrium (R,r) that is not subgame perfect. Furthermore, perfect equilibria of the RNF are reasonable solutions for the extensive form game: At unreached information sets, one can define suitable choices such that the resulting equilibrium is *weakly sequentially rational* (Reny, 1992; Siniscalchi, 1996). Finally, one can try several (for example, randomly chosen) priors, to test the sensitivity of the computed equilibrium with respect to the starting point. We suggest this as a topic for future research.

Appendix: Degeneracy resolution

In an extensive game, certain equilibria may avoid entire branches of the tree. Then, the behavior of one player in these unreached branches is to some extent arbitrary, like the probability y_d for the sequence d in the set of equilibria (x^3,y^3) above. In particular, one of these equilibria has $y_d = 0$ even though d is optimal, that is, $(w_y)_d = 0$. This is a *degeneracy*, namely, a basis containing a basic variable (either y_d or $(w_y)_d$) which has value zero. It arises due to the structure of the game tree, even for generic payoffs, and applies not only to the sequence form but also to the more redundant normal form of the game.

Degeneracy must be dealt with properly, partly for the following technical reason. The complementary pivoting algorithm terminates if no almost complementary basis is revisited. This is the case if the leaving variable is always unique. If there are two variables that may leave the basis, one of them will stay basic and have value zero after the pivoting step, so the resulting basis is degenerate. Thus, if degeneracy can somehow be avoided, the algorithm will terminate in a finite number of steps.

This is achieved by the well-known *lexicographic method*, adapted by Koller, Megiddo, and von Stengel (1996) for our type of algorithm. Consider the system

$$Iw - Mz - dz_0 = q$$

which is equivalent to $w = q + Mz + dz_0$ in (4.3), where I is the $n \times n$ identity matrix. A basis corresponds to an invertible $n \times n$ submatrix C of $[I, -M, -d]$, so that the vector of basic variables is $C^{-1}q$. An infinitesimal perturbance of q, replacing q by $q(\varepsilon) = q +$

$(\varepsilon, \varepsilon^2, \ldots, \varepsilon^n)^\top$ for some positive but vanishingly small ε, then defines a vector $C^{-1}q(\varepsilon)$ of basic variables. These are all positive (although some of them may be vanishingly small) iff the matrix $[C^{-1}q, C^{-1}]$ is lexicographically positive, that is, the first nonzero entry of each row is positive. The basis is then called *lexico-positive*. The invariant that all computed bases are lexico-positive is preserved by pivoting with the "lexico-minimum ratio test", which determines the leaving variable uniquely. The actual values of the basic variables are still $C^{-1}q$, so the computation is unchanged.

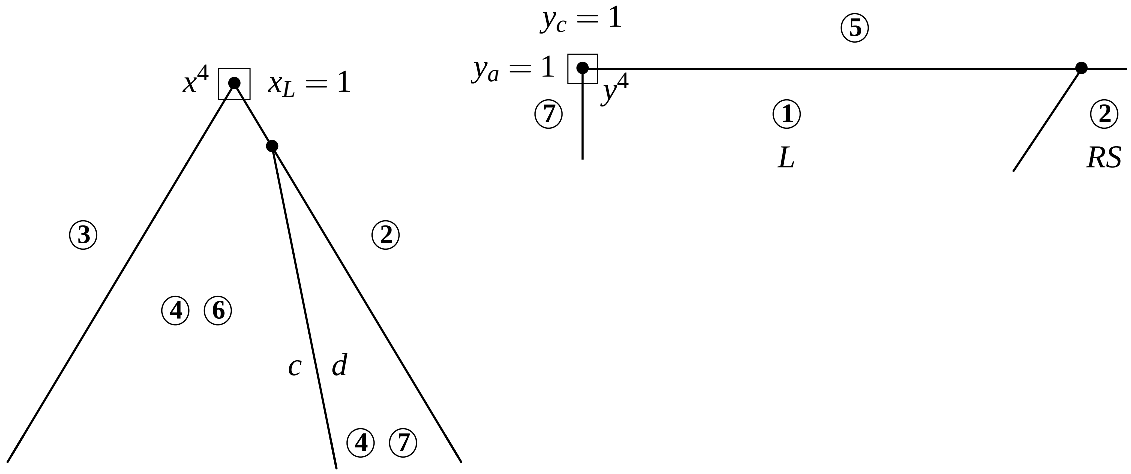

Figure A.1. Partial view of the strategy spaces after artificial perturbance of the LCP data to achieve nondegeneracy. The labels show that only one equilibrium (x^4, y^4) marked with $\square$ remains from the corresponding infinite set of equilibria in Figures 3.3 and 3.4.

Applied to our example, this (simulated) perturbation of the LCP data, if it actually took place with $\varepsilon > 0$, would remove the degeneracies in Figure 3.3 and reduce the infinite set of equilibria (x^3, y^3) marked by boxes in Figures 3.3 and 3.4 to a singleton. Namely, the line in Figure 3.3 separating the best response regions for c and d would no longer hit the vertex $x_L = 1$ but one of the sides of the triangle. Figure A.1 shows one possible effect of this perturbation, where the set of equilibria becomes the single equilibrium (x^4, y^4) in pure strategies with $x^4_L = 1$ and $(y^4_a, y^4_b, y^4_c, y^4_d) = (1, 0, 1, 0)$, as can be seen from the labels of the regions. The other possibility is shown in Figure A.2, where the equilibrium is (x^5, y^5) with x^5_L being nearly one and x^5_{RS} nearly zero, and $(y^5_a, y^5_b, y^5_c, y^5_d) = (1, 0, {}^1\!/_{12}, {}^{11}\!/_{12})$. Above, we have considered the case $\varepsilon = 0$, where $x^3 = x^4 = x^5$ and any convex combination y^3 of y^4 and y^5 is an equilibrium strategy of player 2.

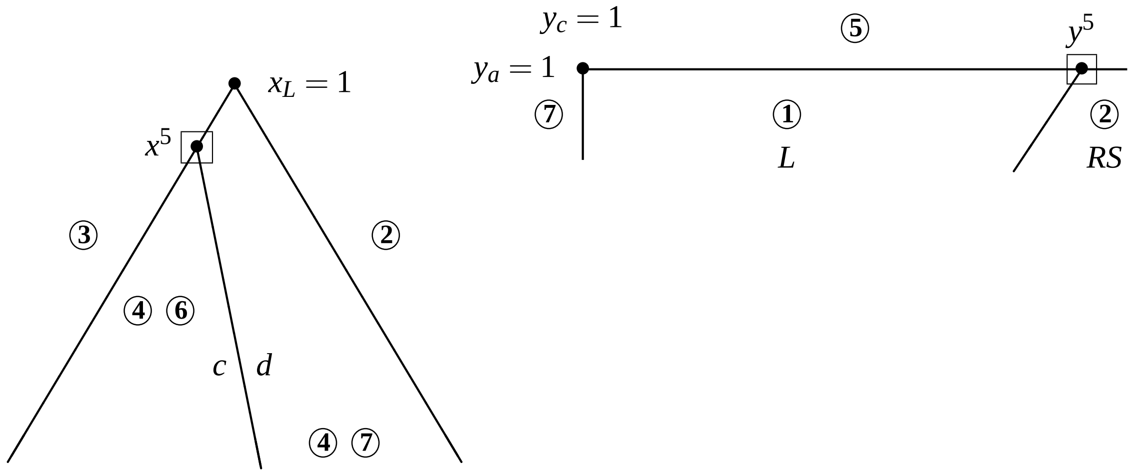

Figure A.2. Similar to Figure A.1, a different perturbance leaves the other extreme equilibrium (x^5, y^5) in the formerly infinite set of equilibria.

For the particular order of LCP variables $x_L, x_{RS}, x_{RT}, y_a, y_b, y_c, y_d$ that we chose in this example, the lexicographic rule corresponds to the situation in Figure A.2 (where x_5 is indistinguishable from the vertex $x_L = 1$). In the final computation step shown in Figure 5.1, the path reaches the degenerate vertex $x_L = 1$ and the algorithm terminates since $z_0 = 0$, according to 4.3(c). Applied to the perturbed problem in Figure A.2, it in fact hits first the best response region for sequence d (labeled ⑦ in Figure A.2), so that another pivoting step happens: $(w_y)_d$ leaves and y_d enters the basis, which is increased while $z_0 = 0$, until the best response region for RS is reached (for $y_d = {}^{11}/_{12}$), whereupon x_{RS} enters the basis, but stays at value zero, and z_0 finally leaves. That is, the algorithm, using the lexicographic rule alone, finds the above equilibrium (x^5, y^5) rather than the pure strategy equilibrium (x^4, y^4). The final degenerate pivoting step is interesting for the following reason. Koller, Megiddo, and von Stengel (1996) have shown that the algorithm, using the lexicographic rule until z_0 leaves the basis, terminates in an equilibrium. It may happen that the variable z_0 could leave the basis earlier because it is zero, but that z_0 is not chosen by the tie-breaking lexicographic rule. So far, it was open whether the extra test if z_0 can leave the basis can shorten the computation. The example shows that such a shortcut is indeed possible. On the other hand, one may explicitly not take this shortcut and note that there is an equilibrium, but keep on computing to find a possibly different equilibrium, here (x^5, y^5). In Algorithm 4.3, condition (c) prevents this.

The *initial* almost complementary basis should also be lexico-positive. This basis, for $z_0 = 1$, is highly degenerate since it has many basic variables with value zero, namely the components of x and y which are basic variables. Which variables should be basic, and which should be the first entering variable? In order to solve this problem, Lemke's algorithm for the standard (not the mixed) LCP can be used from the very beginning, as described in the remainder of this section. This avoids altogether the initialization (a) in Algorithm 4.3 by linear programming. However, this is merely a computationally convenient simplification, and does not affect the interpretation of the algorithm given in the text, including the initial step. We modify our problem as follows.

Assumption A.1. *Every leaf has negative payoffs. The starting vector (s,t) is completely mixed, that is, it assigns positive probability to every sequence σ.*

Negative payoffs can be achieved by subtracting a suitable constant, which does not alter the game. Then, the sparse payoff matrices fulfill $A \leq \mathbf{0}$ and $B \leq \mathbf{0}$. This implies that u and v are never positive in any optimal solutions to the dual LPs with constraints (3.1) and (3.3), respectively, due to the structure of E and F. Thus, we can replace u by $-u$ and require that this new vector u is nonnegative. Then the problem of maximizing the expected payoff $x^\top(Ay)$ has the dual LP: minimize $-e^\top u$ subject to $-E^\top u \geq Ay$, $u \geq \mathbf{0}$. The primal LP, in turn, is to maximize $x^\top(Ay)$ subject to $Ex \geq e$, $x \geq \mathbf{0}$. That is, instead of converting $Ex = e$ to the pair of inequalities $Ex \geq e$ and $Ex \leq e$, we only use the first of these inequalities. The same is done for the second player.

This yields an LCP in standard inequality form without increasing its dimension. The new LCP data are

$$q = \begin{pmatrix} -e \\ -f \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \qquad M = \begin{pmatrix} & & E & \\ & & & F \\ -E^\top & & & -A \\ & -F^\top & -B^\top & \end{pmatrix}. \tag{A.1}$$

The LCP variables $z = (u,v,x,y)^\top$ and $w = (w_u, w_v, w_x, w_y)^\top$ are all nonnegative.

Since $A \leq \mathbf{0}$ and $B \leq \mathbf{0}$, the covering vector d given by (4.5) is nonnegative and has positive components d_i whenever $q_i < 0$. We can therefore use the original initialization of Lemke's algorithm: Let w be the first vector of basic variables. Determine the smallest

z_0 such that $q + dz_0 \geq \mathbf{0}$. This is $z_0 = 1$, where all components of w_u and w_v become zero. With z_0 as entering variable, a component of w is chosen to leave the basis. Using the lexicographic rule, all bases are lexico-positive from then on. The components of w_u and w_v, and certain components of w_x and w_y, are then successively replaced as basic variables by components of u and v and x and y, respectively. These initial pivots are all degenerate, changing the basis but not the values of the basic variables, until $z_0 < 1$. The computation proceeds as before if we can guarantee that the slacks in w_u and w_v never become positive.

Lemma A.2. *Under Assumption A.1, consider the LCP (4.1) with q, M as in (A.1) and covering vector d as in (4.5). Then Lemke's algorithm, using the lexicographic rule and letting z_0 leave the basis as soon as z_0 becomes zero, solves (4.1) and finds an equilibrium (x, y). No component of w_u or w_v assumes a positive value during the computation.*

Proof. By Theorem 4.4 of Koller, Megiddo, and von Stengel (1996), the algorithm terminates, since, among other things, the matrix M is *copositive*, that is, $z \geq \mathbf{0}$ implies $z^\top Mz \geq 0$, which holds because $z^\top Mz = x^\top(-A - B)y$.

Consider an almost complementary basis computed during the algorithm. We are interested in the values of z_0, x, y, w_u, w_v. Assume that, contrary to our claim, there is a proper inequality in $w_u = Ex - e(1 - z_0) \geq \mathbf{0}$ with a positive component of w_u. By (2.3), this means either $x(\emptyset) > 1 - z_0$ or

$$\sum_{c \in C_h} x(\sigma_h c) > x(\sigma_h) \tag{A.2}$$

for some information set h. So for some choice c at h, the component $x(\sigma_h c)$ of x is positive and could be reduced by the value of the slack variable. If the sequence $\sigma_h c$ does not lead to a leaf, it leads to an information set h' following h in the tree and appears as right hand side $x(\sigma_{h'})$ in (A.2). By induction, we finally obtain a sequence σ leading to a leaf where x_σ can be reduced by a positive amount. If $x(\emptyset) > 1 - z_0$, this can also be $\sigma = \emptyset$.

By the complementarity condition, the current x maximizes the linear expression $x^\top A(y + tz_0)$ (with y and z_0 fixed) subject to

$$Ex \geq e(1 - z_0), \qquad x \geq \mathbf{0}. \tag{A.3}$$

However, the vector $A(y+tz_0)$ is nonpositive. We have shown that for a sequence σ leading to a leaf, x_σ can be reduced while preserving (A.3). But this yields a contradiction since the corresponding component $(A(y+tz_0))_\sigma$ is negative. Indeed, we may assume $z_0 > 0$ since this is true for the preceding almost complementary basis (otherwise the algorithm would have terminated) and thus on the last line segment on the computed path (possibly excluding its endpoint). Furthermore, t is completely mixed, so the leaf is reached with positive probability. It suffices that y is nonnegative. Note that $y+tz_0$ is not necessarily a realization plan since $F(y+tz_0) \geq f$ may a priori not hold with equality. Thus, w_u stays zero throughout the algorithm, and so does w_v. $\square$

Note that in Lemma A.2, some components of w_u and w_v may stay basic variables, but they will not assume positive values. Of course, once they are replaced by their complementary components of u or v, respectively, one can as in 4.3(b) make sure that they never enter the basis again. The components of w, in particular of w_u and w_v, correspond to the columns of I in the matrix $[I, -d, -M]$ above. These are useful to keep for the lexicographic rule when pivoting is performed using the *tableau* $C^{-1}[q, I, -d, -M]$ with the basis matrix C.

With a detailed analysis of the tableau during the initial pivoting operations (which is too long to be presented here), Lemma A.2 can be shown under slightly weaker assumptions. In particular, s and t do not have to be completely mixed. However, this assumption is also conceptually useful for computing normal form perfect equilibria, as shown in Section 6.

Acknowledgements

We thank Eric van Damme and Marciano Siniscalchi for helpful comments. B.v.S. was supported by a Heisenberg grant of the Deutsche Forschungsgemeinschaft and thanks CentER for its hospitality during the time of this research.

References

A. Charnes (1953), Constrained games and linear programming. Proc. National Academy of Sciences of the U.S.A. 39, 639–641.

R. W. Cottle, J.-S. Pang, and R. E. Stone (1992), The Linear Complementarity Problem. Academic Press, San Diego.

Y. Dai and A. J. J. Talman (1993), Linear stationary point problems on unbounded polyhedra. Mathematics of Operations Research 18, 635–644.

E. van Damme (1987), Stability and Perfection of Nash Equilibria. Springer, Berlin.

B. C. Eaves (1973), Polymatrix games with joint constraints. SIAM J. Appl. Math. 24, 418–423.

A. H. van den Elzen (1993), Adjustment Processes for Exchange Economies and Noncooperative Games. Lecture Notes in Economics and Mathematical Systems 402, Springer, Berlin.

A. H. van den Elzen and A. J. J. Talman (1991), A procedure for finding Nash equilibria in bimatrix games. ZOR – Methods and Models of Operations Research 35, 27–43.

A. H. van den Elzen and A. J. J. Talman (1995), An algorithmic approach towards the tracing procedure of Harsanyi and Selten. CentER Discussion paper No. 95111, Tilburg.

J. C. Harsanyi and R. Selten (1988), A General Theory of Equilibrium Selection in Games. MIT Press, Cambridge.

K. Kamiya and A. J. J. Talman (1990), Linear stationary point problems. CentER Discussion paper No. 9022, Tilburg.

D. Koller and N. Megiddo (1992), The complexity of two-person zero-sum games in extensive form. Games and Economic Behavior 4, 528–552.

D. Koller and N. Megiddo (1996), Finding mixed strategies with small supports in extensive form games. International Journal of Game Theory 25, 73–92.

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

G. van der Laan (1984), On the existence and approximation of zeroes. Mathematical Programming 28, 1–24.

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

R. D. McKelvey and A. McLennan (1996), Computation of equilibria in finite games. In: Handbook of Computational Economics, Vol. I, eds. H. M. Amman, D. A. Kendrick, and J. Rust, Elsevier, Amsterdam, pp. 87–142.

P. J. Reny (1992), Backward induction, normal form perfection and explicable equilibria. Econometrica 60, 627–649.

J. V. Romanovsky (1962), Reduction of a game with perfect recall to a constrained matrix game (in Russian). Doklady Akademii Nauk SSSR 144, 62–64.

S. H. Schanuel, L. K. Simon, and W. R. Zame (1991), The algebraic geometry of games and the tracing procedure. In: Game Equilibrium Models II: Methods, Morals, and Markets (R. Selten, ed.), pp. 9–42. Springer, Berlin.

A. Schrijver (1986), Theory of Linear and Integer Programming. Wiley, Chichester.

R. Selten (1975), Reexamination of the perfectness concept for equilibrium points in extensive games. International Journal of Game Theory 4, 25–55.

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

M. Siniscalchi (1996), Weak sequential rationality, invariance and normal-form perfection. Manuscript, Stanford University.

B. von Stengel (1996), Efficient computation of behavior strategies. Games and Economic Behavior 14, 220–246.

B. von Stengel (1997), Computing equilibria for two-person games. To appear in the Handbook of Game Theory, Vol. 3, eds. R. J. Aumann and S. Hart, North-Holland, Amsterdam. (Technical Report #253, Dept. of Computer Science, ETH Zurich.)

R. Wilson (1972), Computing equilibria of two-person games from the extensive form. Management Science 18, 448–460.

R. Wilson (1992), Computing simply stable equilibria. Econometrica 60, 1039–1070.