Efficient Computation of Behavior Strategies

Bernhard von Stengel

Institute for Theoretical Computer Science
ETH Zürich
8092 Zürich, Switzerland
email: stengel@inf.ethz.ch

Games and Economic Behavior **14** (1996), 220–246

(page numbers in this manuscript match only approximately)

Abstract. We propose the *sequence form* as a new strategic description for an extensive game with perfect recall. It is similar to the normal form but has linear instead of exponential complexity, and allows a direct representation and efficient computation of behavior strategies. Pure strategies and their mixed strategy probabilities are replaced by sequences of consecutive choices and their realization probabilities. A zero-sum game is solved by a corresponding linear program that has linear size in the size of the game tree. General two-person games are studied in the paper by Koller, Megiddo, and von Stengel in this journal issue.

Journal of Economic Literature Classification Number: C72

Keywords. Behavior strategy, equilibrium, extensive game, linear programming, normal form, reduced normal form.

1. Introduction

In applications, it is often convenient to describe a game in extensive form. The game tree, with its information sets, possible moves, chance probabilities and payoffs, gives a rather complete picture of the situation that is modeled. If the game tree is explicitly given and not generated from certain rules like in a chess game, it is also a data structure of manageable size. On the other hand, the standard way to find optimal strategies for a game in extensive form is very inefficient. Usually, the game is converted to its normal form by considering all pure strategies for each player and the resulting payoffs when these strategies are employed. A pure strategy specifies a move for each information set of the player, so the number of pure strategies is often *exponential* in the size of the extensive game. This holds also for the reduced normal form of an extensive game where pure strategies differing in irrelevant moves are identified. In the case of a two-person zero-sum game, optimal mixed strategies can then be found by linear programming, but the vast increase in the description can make the problem computationally intractable and may even force the analyst to abandon the game theoretic approach altogether (Lucas 1972, p. P-9).

In this paper, we present a computational method without these disadvantages. It is based on a new strategic description of an extensive game, called the *sequence form*. Based on the sequence form, equilibria of the extensive game can be determined by essentially the same algorithms that are known for the normal form. In particular, a two-person zero-sum game can be solved by a linear optimization method like the simplex algorithm. For a two-person game with general payoffs, an equilibrium is found by Lemke's (1965) variant of the complementary pivoting algorithm by Lemke and Howson (1964) for bimatrix games; this is the topic of the paper by Koller, Megiddo, and von Stengel (1995) in this journal issue. (A summary of this and other results is presented in Koller, Megiddo, and von Stengel 1994; that paper also contains the main results of the present text, which has appeared in preliminary form in von Stengel 1993.) In general, these algorithms are *exponentially faster* than with the standard approach because the size of the sequence form is *linear* and not exponential in the size of the game tree.

The sequence form is a matrix scheme similar to the normal form but where pure strategies are replaced by sequences of consecutive moves. Instead of mixed strategy probabilities, we consider the realization probabilities by which these sequences are played. These are nonnegative real numbers that can be characterized by certain linear equations which correspond naturally to the information sets of the player, provided the player has *perfect recall*, that is, his information sets reflect that he does not forget what he knew or did earlier. From the realization probabilities for the sequences one can reconstruct a *behavior strategy* which defines a local randomization at each information set rather than a global randomization over all pure strategies. This construction is similar to the theorem of Kuhn (1953) stating that in a game with perfect recall, any mixed strategy can be replaced by a behavior strategy.

A player can play the game optimally by appropriately choosing the realization probabilities for his sequences. His expected payoff is *linear* in these variables. This is their key advantage over behavior strategy probabilities: The latter are also small in number, they can be characterized by linear equations (as any probabilities), but the expected payoff usually involves products of behavior strategy probabilities. Using the resulting polynomials for computing equilibria is theoretically and practically much more difficult than the approach taken here: Assume a player seeks a best response against fixed strategies of the other players, so he maximizes his payoff. In terms of sequence form variables, this defines a linear program (LP). In the *dual* of this LP, the variables are separated from the strategic variables of the opponent, so that if these are no longer fixed, the constraints are still linear for a two-person game. We thus obtain an optimization problem whose solutions are the equilibria of the game. This problem is a linear program if the game is zero-sum, a linear complementarity problem for a two-person game with general payoffs, and a multilinear optimization problem for an N-person game. Using a sparse representation of the payoffs and constraints, the optimization problem has a size proportional to the size of the extensive game.

The first polynomial-time algorithm for solving a zero-sum game in extensive form with perfect recall was described by Koller and Megiddo (1992). It solves a linear program with essentially the same variables as in our approach. The LP inequalities are defined by strategies of the opponent, which may be exponential in number. However, these inequalities can be evaluated as needed by finding a best response of the opponent, which can be done quickly by backward induction. This solves efficiently the 'separation problem' for the ellipsoid method for linear programming, which therefore runs in polynomial time. Similarly, Wilson (1972) described a method for solving extensive two-person games with general payoffs, where best responses, which serve as pivoting columns for the Lemke-Howson algorithm, are generated directly from the game tree. In contrast to these approaches, we no longer consider pure strategies but use sequences symmetrically for all players. In our LP, the number of variables *and* constraints is linear in the size of the game tree. We will compare our techniques in detail with earlier work in the concluding Section 7.

In Section 2, we state our notation and basic definitions, and introduce a simple example that will be used frequently. We define the sequence form in Section 3. The strategic variables of the players describe how sequences are played, and can be translated to behavior strategies. In Section 4, we consider mixed strategies and compare the sequence form with the well-known reduced normal form of an extensive game. In Section 5, we apply the sequence form to two-person games. A central idea is the linear program for computing a best response of one player to fixed strategies of the other players. From this one obtains a linear program whose solutions are the optimal strategies of a zero-sum game, and a linear complementarity problem whose solutions are the equilibria of a non-zero-sum game. In Section 6, we interpret the dual solutions of the 'best response LP' and describe the N-person case. In Section 7, we summarize our results, compare them with earlier work, and discuss their applicability to games without perfect recall.

As prerequisites, we assume familiarity with the duality of linear programs. Classical texts are Gale (1960) and Dantzig (1963). A more recent introduction to linear programming is Chvátal (1983).

2. Extensive and Normal Form Games

In this section, we state our notation and conventions for games in extensive and normal form. For an extensive game, it will be convenient to represent choices by unique labels of edges in the game tree, and to treat the random chance moves as a fixed behavior strategy played by a chance player. We also give an example that will be used throughout the text.

The basic structure of an extensive game is the *game tree*, which is a finite, directed tree, that is, a directed graph with a distinguished node, the *root* (initial node), from which there is a unique path to every other node. Edges of the tree are denoted by ab, where the node b is called a *child* of the node a. Nodes without children (that is, terminal nodes) are called *leaves*, the others *decision nodes*. Trees are depicted graphically with the root at the top and edges going downwards.

In addition to the game tree, an extensive game has the following components. In general, there are N *personal players* of the game numbered $1, \ldots, N$. An additional *chance* player is denoted as player 0. The chance player is here treated symmetrically to the other players, except that he plays with a fixed behavior strategy and receives no payoff.

The *payoff* function h is defined on the set of leaves and yields a vector $h(a)$ in $\mathbb{R}^N$ for each leaf a. The ith component $h_i(a)$, $1 \leq i \leq N$, of $h(a)$ is called the payoff to player i at a. A *zero-sum* game has two players ($N = 2$), with $h_2 = -h_1$.

The possible moves of a player are represented by a function that assigns to each edge ab a label, called a *choice* at a, such that the choices at a are always distinct, that is, the children b of a decision node a can be distinguished by the respective labels of the edges ab. The usage of the term *move* varies in the literature. We use it to denote an action that occurs during play, whereas a choice means an intended move as planned by a player, or a possible move at a decision node of the game tree. There is not much harm in confusing these terms since both refer to the outgoing edges at decision nodes. Similarly, a *play* is a particular instance in which the game is played from beginning to end, that is, a sequence of moves represented by the path from the root to a particular leaf of the tree. A *game* is the static description with the entire tree and all possible moves and outcomes.

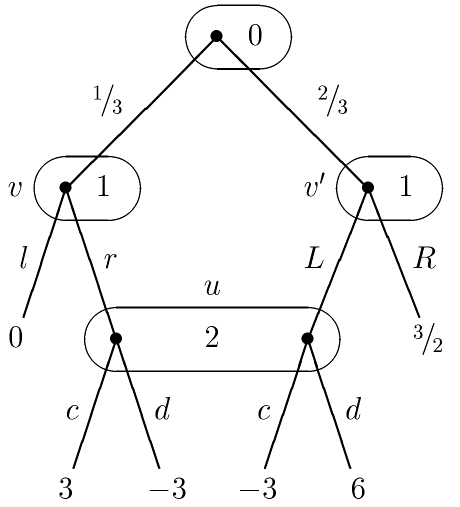
The set of decision nodes is partitioned into *information sets.* Each information set u belongs to exactly one player i, $0 \le i \le N$, called the *player to move* at u. The set of all information sets of player i is denoted by U_i. For all nodes a in u there are the same choices at a, which will be called the choices at u; the set of these choices is denoted by C_u. In particular, all nodes a in u have the same number $|C_u|$ of children. For simplicity, it is assumed that the choice sets C_u and C_v of any two information sets u and v are disjoint. The set $\bigcup_{u \in U_i} C_u$ of all choices of player i, $0 \le i \le N$, is denoted D_i.

Finally, fixed positive probabilities for the chance moves are also part of the extensive game. They are specified as a behavior strategy β_0 for player 0. Generally, a *behavior strategy* β_i for player i is given by a probability distribution on C_u, called the *behavior* at u, for each information set $u \in U_i$. It is represented as a function $\beta_i : D_i \to \mathbb{R}$ such that the probability $\beta_i(c)$ for making the move c is nonnegative for all $c \in D_i$, and $\sum_{c \in C_u} \beta_i(c) = 1$ for all $u \in U_i$. Without loss of generality, the chance probabilities $\beta_0(c)$ for $c \in D_0$ are assumed to be nonzero, since branches of the game tree that are never reached can be pruned.

In the extensive games considered here, each player is assumed to have *perfect recall.* This is a structural property of the information sets of the player, reflecting that he does not forget what he knew or did earlier. Technically, it says that a choice c at an information set u of a player precedes a node b of an information set v of the same player if and only if it precedes all nodes of v. Thereby, c is said to *precede* b if c is the label of an edge on the path from the root to b. The concept of perfect recall has been introduced by Kuhn (1953). This short definition is due to Selten (1975); see also Wilson (1972, p. 456). It is easy to see that perfect recall implies that there is no path between two nodes of an information set, a property which often forms part of the definition of an extensive game.

An example of an extensive game is shown in Fig. 2.1. There is a chance move from the root with probability $1/3$ to the left and with probability $2/3$ to the right. The labels in D_0 denoting the choices of the chance player are omitted since the chance probabilities suffice, but in general they will be useful for systematic reasons. There are three information sets u, v, v' indicated by ovals. The game is zero-sum, so only the payoffs to player 1 are specified at the leaves of the tree. For two-person games, we will use the pronoun 'she' for the first player and 'he' for the second player.

In the game in Fig. 2.1, both players have perfect recall. This would not be the case if the first information set belonged to player 2 and not to the chance player. Games without perfect recall and the associated computational difficulties are discussed in Section 7 below.



	c	d
(l, L)	-2	4
(l, R)	1	1
(r, L)	-1	3
(r, R)	2	0

Fig. 2.1. An extensive game, and its normal form.

For a general extensive game, *pure strategy* π_i of player i, $1 \le i \le N$, specifies a choice at each information set $u \in U_i$, so the set P_i of his pure strategies is the cartesian product $\prod_{u \in U_i} C_u$. For each N-tuple $\pi = (\pi_1, \ldots, \pi_N)$ of pure strategies, the payoff vector $H(\pi) \in \mathbb{R}^N$ is given by the expected payoff that results from the payoffs $h(a)$ at the leaves a reached by the random chance moves and the players' moves as prescribed by π. This defines a game in *normal form.* In general, such a game is given by $(P_1, \ldots, P_N; H)$ with $H\colon P_1 \times \cdots \times P_N \to \mathbb{R}^N$, where $P_1, \ldots, P_N$ are non-empty finite sets of pure strategies.

A *mixed strategy* μ_i of player i is a probability distribution on his set P_i of pure strategies. For a tuple $\mu = (\mu_1, \ldots, \mu_N)$ of mixed strategies, the expected payoff $H(\mu)$ is defined accordingly. Such a tuple μ is an *equilibrium* if each mixed strategy μ_i, $1 \le i \le N$, is a *best response* to the fixed $(N-1)$-tuple of the mixed strategies of the other players, that is, yields maximum payoff $H_i(\mu)$ to player i for all his mixed strategies.

Figure 2.1 also shows the normal form of the extensive game. Each row represents a pure strategy of player 1 and each column a pure strategy of player 2. The respective matrix entry is the expected payoff to player 1 when these strategies are used. The normal form of an extensive game may be very large since the number of pure strategies is exponential in the number of information sets. For example, in an extensive game similar to Fig. 2.1 but with n 'parallel' information sets, instead of only two, following an initial chance move with n possibilities, and with two choices at each of these information sets, player 1 has 2^n strategies.

One equilibrium of this game consists of the pure strategy (l, R) for player 1 and the mixed strategy for player 2 that assigns probability $1/2$ to both c and d. These strategies are mutual best responses. A mixed strategy is a best response to a mixed strategy of the other player if and only if every pure strategy selected with positive probability is a best response (Nash 1951, p. 287). Player 2 can therefore assign positive probabilities to c and d since both give him the same maximum expected payoff -1. Conversely, against the mixed strategy $(1/2, 1/2)$ of player 2, all pure strategies of player 1 are best responses, which suggests that there may be further equilibria. Indeed, there exist two mixed strategies μ_1 and μ_1' of player 1 that produce the same expected payoffs to player 2. These are, as vectors of probabilities for her four pure strategies, $(0, 0, 1/3, 2/3)$ and $(1/4, 0, 0, 3/4)$, in addition to the pure strategy $(0, 1, 0, 0)$. Since the game is zero-sum, all convex combinations of these three extreme mixed strategies are also equilibrium strategies.

3. The Sequence Form

In this section, we define the sequence form for an extensive N-person game, using the notation of the previous section. The sequence form is a new strategic description that describes strategies in a new way: Rather than planning a move for every information set, a player looks at each leaf of the game tree and considers the choices he needs to make so that the leaf can be reached in the game. These choices are prescribed by the respective play, i.e. path from the root to the leaf, whenever that path goes through an information set of the player. They represent a 'sequence' that will be considered instead of a pure strategy:

Definition 3.1. A *sequence* of choices of player i, $0 \le i \le N$, *defined* by a node a of the game tree, is the set of labels in D_i on the path from the root to a. The set of sequences of player i is denoted S_i.

A sequence can also be regarded as the string of choice labels of the player found on the path to some node a. For easy reference to its elements, we have defined a sequence as a set. This is possible since choices at different information sets are distinct. In Fig. 2.1, for example, player 1 has the sequences (represented as strings) l, r, L and R, and the empty sequence $\emptyset$; the sequences of player 2 are $\emptyset$, c and d.

In the sequence form, S_i is the set of sequences of player i defined by all nodes of the game tree. It replaces his set of pure strategies in the normal form. The sequences of the chance player 0 are also considered, since this allows dealing only with payoffs and not expected payoffs. As in the normal form, payoffs are defined as they result from combinations of sequences:

Definition 3.2. The *payoff function* $g: S_0 \times S_1 \times \cdots \times S_N \to \mathbb{R}^N$ is defined by $g(s) = h(a)$ if s is the $(N+1)$-tuple $(s_0, s_1, \ldots, s_N)$ of sequences defined by a leaf a of the game tree, and by $g(s) = (0, \ldots, 0) \in \mathbb{R}^N$ otherwise.

The payoff function is well defined since the $(N+1)$-tuple $(s_0, s_1, \ldots, s_N)$ of sequences, where s_i is defined by a for $0 \leq i \leq N$, is unique for any node a of the game tree. For a player, there are at most as many sequences as the game tree has nodes, so their number is linear in the size of the game tree, in contrast to the number of pure strategies which may be exponential. The ith component g_i of the payoff function g for $1 \leq i \leq N$ is a $(1+N)$-dimensional matrix representing the payoffs to player i. That matrix is sparse since the number of nonzero entries is at most the number of leaves of the tree, so the size of the matrix is also linear if it is represented sparsely. In contrast, the payoff matrix for a player in the normal form is usually full. Because the chance probabilities are known, the dimension of the matrix can be reduced from $1+N$ to N by considering expected payoffs, as shown in Section 5.

In addition to the payoffs, it is also necessary to specify how sequences are selected by a player. In the normal form, the player may just decide on a pure strategy, or, by a mixed strategy, use a probability distribution to select one. In the sequence form, a player cannot just decide on a single sequence. In Fig. 2.1, for example, player 1 has to decide between l and r as well as between L and R, so she may for example decide on l and L as in the pure strategy (l, L). In that case, the probabilities $1, 1, 0, 1, 0$ are assigned to her sequences $\emptyset, l, r, L, R$, respectively. In general, mixed strategy probabilities will be replaced by the realization probabilities of sequences when the player uses a *behavior* strategy. The use of a mixed strategy will be considered in the next section.

If player i, $0 \leq i \leq N$, uses the behavior strategy β_i, then the sequence $s_i \in S_i$ is played with probability

$$r_i(s_i) = \prod_{c \in s_i} \beta_i(c). \tag{3.1}$$

The function $r_i : S_i \to \mathbb{R}$ defined in this way is called the *realization plan* of β_i.

Realization plans of behavior strategies can be characterized by certain linear equations, using a correspondence between the sequences and the information sets of a player. By definition of perfect recall, every node in an information set u of player i defines the same sequence of choices for that player. This sequence will be denoted σ_u and is called the sequence *leading to* u. A choice c at u extends σ_u. The extended sequence will be abbreviated as $\sigma_u c$,

$$\sigma_u c = \sigma_u \cup \{ c \} \qquad \text{for } c \in C_u.$$

This shows that a nonempty sequence is uniquely specified by its last choice c. Thus, the set S_i of sequences of player i can be represented as

$$S_i = \{ \emptyset \} \ \cup \ \{ \sigma_u c \mid u \in U_i, \ c \in C_u \}. \tag{3.2}$$

Therefore, S_i has $1 + |D_i|$, that is, $1 + \sum_{u \in U_i} |C_u|$ elements.

If one were only interested in the payoffs, it would suffice to consider only the sequences defined by the leaves a of the game tree. By considering decision nodes a as well, the representation (3.2) of S_i leads to the following constraints that are fulfilled by a realization plan r_i according to (3.1):

$$r_i(\emptyset) = 1 \tag{3.3}$$

because the empty product is 1, and, since $\sum_{c \in C_u} \beta_i(c) = 1$,

$$-r_i(\sigma_u) + \sum_{c \in C_u} r_i(\sigma_u c) = 0 \qquad \text{for } u \in U_i. \tag{3.4}$$

Furthermore, realization probabilities are obviously nonnegative:

$$r_i(s_i) \geq 0 \qquad \text{for } s_i \in S_i. \tag{3.5}$$

The following definition of a realization plan uses these constraints; it is justified by the subsequent proposition.

Definition 3.3. A function $r_i : S_i \to \mathbb{R}$ fulfilling (3.3), (3.4) and (3.5) is called a *realization plan* for player i, $0 \leq i \leq N$.

Proposition 3.4. Any realization plan arises from a suitable behavior strategy.

Proof. The realization plan r_i arises from the following behavior strategy β_i. For an information set u in U_i, define the behavior at u by

$$\beta_i(c) = \frac{r_i(\sigma_u c)}{r_i(\sigma_u)} \qquad \text{for } c \in C_u \tag{3.6}$$

if $r_i(\sigma_u) > 0$, and arbitrarily (so that $\sum_{c \in C_u} \beta_i(c) = 1$) if $r_i(\sigma_u) = 0$. Then (3.1) follows by induction on the length of a sequence. $\square$

Definition 3.5. The *sequence form* of an extensive game is given by the sets of sequences, the payoff function g, the constraints for the realization plans of the personal players, and the realization plan r_0 of the given behavior strategy β_0 of the chance player.

The sequence form corresponds closely to the extensive game. It is an abstraction like the normal form that describes the strategic possibilities of the players and the resulting payoffs. It has the advantage of small size and the disadvantage of a less intuitive selection of sequences by realization plans. The latter are finitely described by the constraints in Def. 3.3.

The constraints for player i, $1 \leq i \leq N$, are determined if one knows for each information set u in U_i the sequence σ_u leading to u and the choices c at u. They need not be stated for the chance player because r_0 is defined by β_0. It is possible to ignore the structure of sequences and regard S_i just as a finite set, and a realization plan r_i as a nonnegative vector with $|S_i|$ components. The linear equations (3.3) and (3.4) for this vector can then be represented by a two-dimensional 'constraint' matrix with $1 + |U_i|$ rows and $|S_i|$ columns. We will do this in Section 5 where we show, using matrix notation, how equilibria of two-person games can be computed with the sequence form.

In the normal form, the *expected* payoff to each player is a multilinear expression in terms of the payoffs and the mixed strategy probabilities. In the sequence form, it is defined analogously. Consider behavior strategies $\beta_1, \ldots, \beta_N$ for all players, and let $r = (r_1, \ldots, r_N)$ be the tuple of corresponding realization plans. As in Def. 3.5, let r_0 be the fixed realization plan for the chance player. Let $S = S_0 \times S_1 \times \cdots \times S_N$ denote the set of all $(N+1)$-tuples $s = (s_0, s_1, \ldots, s_N)$ of sequences. Then, define the expected payoff vector $G(r) \in \mathbb{R}^N$ in terms of $g(s)$ for $s \in S$ by

$$G(r) = \sum_{s \in S} g(s) \prod_{i=0}^{N} r_i(s_i). \tag{3.7}$$

Indeed, $G(r)$ is the expected payoff $H(\beta_1, \ldots, \beta_N)$ if the players use the behavior strategies β_i because, in the summation over S, only a tuple s of sequences defined by a leaf a of the game tree contributes a nonzero payoff vector $g(s) = h(a)$, and by (3.1), a is reached with probability $\prod_{i=0}^{N} r_i(s_i)$.

4. Mixed Strategies and the Reduced Normal Form

So far, we have defined a realization plan only for a behavior strategy. A mixed strategy can also be strategically represented by a realization plan, as follows. A pure strategy is a special behavior strategy that has a realization plan with integral values zero or one. A mixed strategy μ_i is a convex combination of pure strategies. The corresponding convex combination of the realization plans of pure strategies is again a realization plan by Def. 3.3, and defines the realization plan of μ_i. (This means that the set of realization plans is a polytope, analogous to the simplex of mixed strategies, but of much smaller dimension. The vertices of the polytope represent the pure strategies.) Equivalently, the realization plan of μ_i assigns to each sequence s_i the combined probability under μ_i for the pure strategies that are 'consistent' with s_i.

In going from a mixed strategy to its realization plan, information is lost because the latter has much fewer components. However, the strategically relevant aspect of a mixed strategy is captured by its realization plan: Two mixed strategies of a player are called *realization equivalent* if for any fixed strategies of the other players, both strategies define the same probabilities for reaching the nodes of the game tree. Looking at these probabilities, it is easy to see the following (compare also Koller and Megiddo 1995, Lemma 2.5):

Proposition 4.1. Mixed strategies are realization equivalent if and only if they have the same realization plan.

(For the 'only if' part, any node of the game tree must be reachable for suitable strategies of the other players, which requires that all chance moves have positive probability.) Since a behavior strategy is in effect a special mixed strategy, Propositions 3.4 and 4.1 imply Kuhn's theorem (1953, p. 214):

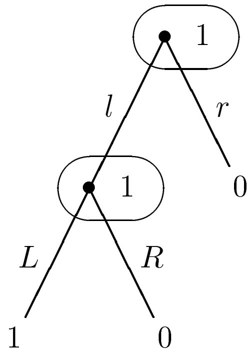
Corollary 4.2. For a player with perfect recall, any mixed strategy is realization equivalent to a behavior strategy.

This shows that for a game with perfect recall, the sequence form, which is designed to compute behavior strategies, is not more restrictive than the normal form. The realization plan of a mixed strategy μ_i can be retranslated to a behavior strategy that is realization equivalent to μ_i.

As an example, consider the optimal mixed strategies of player 1 in Fig. 2.1 described at the end of Section 2. They are $(l, R) = (0, 1, 0, 0)$ and the mixed strategies $\mu_1 = (0, 0, 1/3, 2/3)$ and $\mu_1' = (1/4, 0, 0, 3/4)$. Denote a realization plan by its vector of values for the sequences $\emptyset, l, r, L, R$. The realization plan of (l, R) is $(1, 1, 0, 0, 1)$. For μ_1, selecting (r, L) and (r, R) with probabilities $1/3$ and $2/3$, respectively, it is $1/3 \cdot (1, 0, 1, 1, 0) + 2/3 \cdot (1, 0, 1, 0, 1) = (1, 0, 1, 1/3, 2/3)$. The pure strategy (l, R) as well as the mixed strategy μ_1 are in effect behavior strategies. This is not the case for the optimal mixed strategy μ_1', selecting (l, L) and (r, R) with probabilities $1/4$ and $3/4$, which defines the realization plan $(1, 1/4, 3/4, 1/4, 3/4)$. That realization plan tells that with μ_1', player 1 moves left or right at both information sets v and v' with probability $1/4$ and $3/4$, respectively, but without the correlation of these moves as specified by the mixed strategy μ_1'. Furthermore, this vector is a convex combination of other optimal realization plans, $(1, 1, 0, 0, 1)$ and $(1, 0, 1, 1/3, 2/3)$, unlike μ_1' in the normal form.

More than one behavior strategy β_i may define the same realization plan r_i. As shown in the proof of Prop. 3.4, this is the case if the information set u of player i cannot be reached when β_i is played, that is, if $r_i(\sigma_u) = 0$. Then u is called *irrelevant* when playing β_i, otherwise *relevant* (Kuhn 1953, p. 201). In this case, the behavior at u under β_i is arbitrary and therefore not unique (if u has at least two choices).

In particular, more than one pure strategy may define the same realization plan. However, there is a natural one-to-one correspondence between realization plans with integral values zero or one and pure strategies in the *reduced normal form* of the extensive game. In the reduced normal form, any two pure strategies that differ only in choices at irrelevant information sets are identified, like the strategies (r,L) and (r,R) in Fig. 4.1. Kuhn (1953, p. 202) called such strategies 'equivalent', and even identified them directly (1950, p. 574). They can be represented in the reduced normal form by leaving the choices at the irrelevant information sets blank, denoted by some new symbol like '$*$' that does not denote a choice, like in $(r,*)$ in the example.



(l,L)	1
(l,R)	0
(r,L)	0
(r,R)	0

(l,L)	1
(l,R)	0
$(r,*)$	0

Fig. 4.1. An extensive game, its normal form, and its reduced normal form.

In the reduced normal form, precisely the pure strategies that are realization equivalent are identified. These lead to the same payoffs for all players. However, their identification in the reduced normal form does not depend on the particular payoff function. In contrast, Dalkey (1953, p. 222) defined the reduced normal form via payoff equivalence, which may allow the identification of further strategies like (l,R) and $(r,*)$ in Fig. 4.1. Other reductions of the normal form have been considered by Swinkels (1989), that for a 'generic' game is the reduced normal form considered here.

The reduced normal form can easily be constructed directly from the game tree, without considering the full normal form first. It is smaller than the normal form, but not necessarily in the same significant way as the sequence form:

Definition 4.3. Two information sets u and v of a player are called *parallel* if $\sigma_u = \sigma_v$.

An example of parallel information sets are v and v' in Fig. 2.1. Parallel information sets are not distinguished by preceding choices of the same player, so all combinations of choices at these sets are part of separate pure strategies in the reduced normal form. Because there may be arbitrarily many parallel information sets, the reduced normal form may still be exponentially large. Only if every sequence σ_u leading to an information set u identifies the information set uniquely (like in Fig. 4.1), then the reduced normal form is as compact as the sequence form.

5. Computing Equilibria of Two-Person Games

The sequence form leads to an optimization problem whose solutions are the equilibria of the game. In this section, we derive this problem for the case of a two-person game ($N = 2$). The variables of this problem are the realization plans of the players. We consider first the LP where a best response is sought for one player, and its dual LP. With the help of these linear programs, we can formulate an LP whose solutions are the equilibria of a zero-sum game. For a two-person game with general payoffs, we obtain in a similar way a linear complementarity problem (LCP), which is studied in detail in Koller, Megiddo, and von Stengel (1995).

In this section, we consider an extensive game with two personal players that have perfect recall. The game is transformed to its sequence form. We are looking for a pair of realization plans that represents an equilibrium. As described above, realization plans are easily converted to behavior strategies. In that sense, a player is said to play according to a realization plan. The components of the realization plans, analogous to the mixed strategy probabilities in the normal form, are the strategic variables of the players.

We will use a notation with vectors and matrices. The realization plans r_1 and r_2 for player 1 and 2 shall be written as vectors x and y with $|S_1|$ and $|S_2|$ components, respectively. All vectors are column vectors, row vectors are denoted by transposition as in x^T. The *constraint matrices* E and F are used to express that x and y are realization plans according to Def. 3.3, with the equations

$$Ex = e \qquad \text{and} \qquad Fy = f. \tag{5.1}$$

The number of columns of E and F is $|S_1|$ and $|S_2|$, respectively. The number of rows of E and e is $1 + |U_1|$, and for F and f it is $1 + |U_2|$. The vectors e and f are equal to the unit vector $(1, 0, \ldots, 0)^T$ of appropriate dimension. The first row in each equation in (5.1) represents (3.3), the other rows (3.4).

For the example of Fig. 2.1, these constraint matrices have the following form. The sequences of player 1 are $\emptyset, l, r, L, R$, and for player 2 they are $\emptyset, c, d$. The constraint matrices are

$$E = \begin{pmatrix} 1 & & & & \\ -1 & 1 & 1 & & \\ -1 & & & 1 & 1 \end{pmatrix} \qquad \text{and} \qquad F = \begin{pmatrix} 1 & & \\ -1 & 1 & 1 \end{pmatrix},$$

with $e = (1, 0, 0)^T$ and $f = (1, 0)^T$. The zero entries of the constraint matrices have been omitted. By (3.3) and (3.4), a constraint matrix has a single 1 in every column, and an additional -1 in every row except the first. The constraint matrix is sparse since all other matrix elements are zero.

If player 1 and 2 play according to x and y, respectively, then their expected payoffs can be represented by $x^T Ay$ and $x^T By$, with suitable $|S_1| \times |S_2|$ matrices A

and B. The matrix entry for A in row s_1 and column s_2 is $\sum_{s_0 \in S_0} g_1(s_0, s_1, s_2)\, r_0(s_0)$, and for B it is the same with g_2 in place of g_1. All other matrix elements are zero. The expected payoffs are $x^T Ay$ and $x^T By$ by (3.7).

The payoff matrices A and B are 'flattened' versions of the three-dimensional functions g_1 and g_2, using the known realization plan r_0 for the chance player. The matrices can be constructed directly from the game tree as follows. First, they are initialized as zero. Then, each leaf of the tree is considered, which defines a triple s_0, s_1, s_2 of sequences. The player's payoff at the leaf is multiplied by the product $r_0(s_0)$ of chance probabilities along the path to the leaf. The result is added to the matrix element at position s_1, s_2. This addition is necessary because, due to chance moves, more than one leaf may define the same matrix position. This is done for all leaves. Thus, each payoff matrix has at most as many nonzero elements as the tree has leaves, and it has linear size if it is represented sparsely. In the example of Fig. 2.1, with the sequences $\emptyset, l, r, L, R$ and $\emptyset, c, d$ indicating rows and columns, respectively,

$$A = \begin{pmatrix} & & \\ 0 & & \\ & 1 & -1 \\ & -2 & 4 \\ 1 & & \end{pmatrix};$$

the first row of A is zero (and left blank) since the sequence $\emptyset$ of player 1 is not defined by a leaf; the second row has a zero entry that is shown explicitly because it results from the payoff at a leaf. Because this game is zero-sum, $B = -A$.

In order to derive algorithms for computing an equilibrium, we consider first the problem of finding a best response for one player against a *given* realization plan of the other player. Assume that player 1 plays according to x. Then, finding a best response y of player 2 defines the following linear program:

$$\begin{aligned} \underset{y}{\text{maximize}} \quad & (x^T B)y \\ \text{subject to} \quad & Fy = f\,, \\ & y \geq 0\,. \end{aligned} \tag{5.2}$$

The *dual* LP for this problem will be useful. It has $1+|U_2|$ dual variables represented by the vector q. These variables are unconstrained because they correspond to the equalities in (5.2). The $|S_2|$ constraints for the dual LP correspond to nonnegative primal variables (the components of y), so they are inequalities. Thus, the dual LP to (5.2) is

$$\begin{aligned} \underset{q}{\text{minimize}} \quad & q^T f \\ \text{subject to} \quad & q^T F \geq x^T B\,. \end{aligned} \tag{5.3}$$

Analogously, a best response x of player 1, given that player 2 plays according to y, is a solution to the following problem:

$$\begin{array}{ll} \underset{x}{\text{maximize}} & x^T(Ay) \\ \text{subject to} & x^T E^T = e^T, \\ & x \qquad \geq 0\,. \end{array} \tag{5.4}$$

The dual problem to (5.4) uses the unconstrained vector p with $1+|U_1|$ components and reads

$$\begin{array}{ll} \underset{p}{\text{minimize}} & e^T p \\ \text{subject to} & E^T p \geq Ay\,. \end{array} \tag{5.5}$$

In order to find an equilibrium, both x and y have to be treated as variables. Then, the objective functions in (5.2) and (5.4) are no longer linear. Nevertheless, a zero-sum game can be solved with a linear program. We consider this case first and treat general payoffs later.

In a zero-sum game, we regard the dual LP (5.5), but with variables p and y, based on the following intuition: The LP (5.4) and its dual (5.5) have the same optimal value $e^T p$. This is the payoff that player 2, if he plays y, has to give to player 1. If player 2 can vary y, he will try to minimize this payoff; an optimal choice of y will be a min-max strategy. Thereby, y must be subject to $y \geq 0$ and $Fy = f$ as in (5.1) to represent a realization plan for player 2; it will be convenient to write the equation with a negative sign. This defines the new LP

$$\begin{array}{lrcl} \underset{y,\,p}{\text{minimize}} & e^T p & & \\ \text{subject to} & -Ay + E^T p & \geq & 0 \\ & -Fy & = & -f\,, \\ & y & \geq & 0\,. \end{array} \tag{5.6}$$

Again, consider the dual of this LP:

$$\begin{array}{lrcl} \underset{x,\,q}{\text{maximize}} & -q^T f & & \\ \text{subject to} & x^T(-A) - q^T F & \leq & 0 \\ & x^T E^T & = & e^T, \\ & x & \geq & 0\,. \end{array} \tag{5.7}$$

In a zero-sum game, $-A = B$, so (5.7) is just (5.3) but with variables q and x, subject to $x \geq 0$ and $Ex = e$ as in (5.1). The LP (5.7) can be interpreted as the problem of finding a min-max strategy for player 1. The following result states that the optimal solutions to (5.6) and (5.7) define indeed an equilibrium of the zero-sum game. It is proved with the duality theorem of linear programming.

Theorem 5.1. The equilibria of a zero-sum game in extensive form with perfect recall are the optimal primal and dual solutions of a linear program whose size, in sparse representation, is linear in the size of the game tree.

Proof. The linear program is (5.6). The number of nonzero entries of the payoff matrix A and of the constraint matrices E and F is linear in the size of the game tree. Consider an optimal solution y, p to (5.6) and a dual optimal solution x, q to (5.7). Then, y, q, x and p are feasible solutions to the linear programs (5.2), (5.3), (5.4) and (5.5), respectively, where q and x fulfill the constraints of (5.3) since $B = -A$. Multiplying the equation $f = Fy$ in (5.2) by q^T and the inequality in (5.3) by the nonnegative vector y yields

$$q^T f = q^T F y \geq x^T B y \,. \tag{5.8}$$

Analogously,

$$e^T p = x^T E^T p \geq x^T A y \,, \tag{5.9}$$

which implies

$$e^T p \geq x^T A y = -x^T B y \geq -q^T f \,. \tag{5.10}$$

Condition (5.8) is known as the weak duality theorem, that is, the objective function $x^T B y$ of the LP (5.2) is bounded by the objective function $q^T f$ of the dual LP (5.3) and vice versa, for any pair of feasible solutions. Similarly, (5.9) says this for the pair of linear programs (5.4) and (5.5), and (5.10) states it in effect for (5.6) and (5.7). According to the strong duality theorem, a pair of primal and dual solutions is optimal if and only if the two objective functions are equal. Applied to (5.6) and its dual (5.7), this shows $e^T p = -q^T f$, so equality holds in (5.10), (5.9) and (5.8). Therefore, x is an optimal solution of the LP (5.4) and a best response to y, and analogously y is a best response to x. That is, x, y represents an equilibrium.

Conversely, any equilibrium x, y solves the linear programs (5.4) and (5.2) optimally, and with the corresponding optimal dual solutions p, q, equality holds in (5.9), (5.8) and (5.10), so that (5.6) and (5.7) are solved optimally. □

Theorem 5.1 shows that a zero-sum game in extensive form can be solved in polynomial time, using any polynomial linear programming algorithm. In practice, the LP (5.6) is very suitable for the simplex algorithm, which computes an optimal pair of primal and dual solutions. Efficient implementations exploit the sparsity of the matrices A, E and F (Chvátal 1983, p. 112). The running time of the simplex algorithm can be exponential but is usually quite short. It is mostly determined by the number of constraints and very little by the number of variables (Chvátal 1983, p. 46). Therefore, it may be advantageous in certain cases to run the simplex algorithm on the dual LP (5.7) instead of (5.6). There are $|S_1|+1+|U_2|$ constraints in (5.6) and $|S_2|+1+|U_1|$ constraints in (5.7). Note that both numbers are of order $|U_1| + |U_2|$ unless a player has a large number of choices per information set.

In the case of a non-zero-sum game, we will not consider the LP (5.6). Instead, we will use the complementary slackness condition that characterizes optimal LP solutions. As mentioned in the proof of Theorem 5.1, x and p are optimal solutions to (5.4) and (5.5) if and only if (5.9) holds with equality, that is, if $x^T E^T p = x^T Ay$ or

$$x^T\,(-Ay + E^T p) = 0\,. \tag{5.11}$$

Equation (5.11) states that the nonnegative vector x of primal variables is orthogonal to the nonnegative vector $-Ay + E^T p$ of *slacks* in the dual program (the dual condition $(x^T E^T - e^T)\,p = 0$ holds also but is trivial). In linear programming, this orthogonality condition is known as *complementary slackness*. It characterizes x as a best reponse to y. We will interpret this in the next section.

Similarly, y and q are optimal solutions to (5.2) and (5.3) if and only if (5.8) holds with equality, that is, if $q^T F y = x^T B y$ or

$$(-x^T B + q^T F)\,y = 0\,. \tag{5.12}$$

An equilibrium x, y is given by simultaneous optimal solutions to the linear programs (5.4) and (5.2). Considering the dual programs (5.5) and (5.3) as well, this defines the following problem: Find x, y, p, q so that the constraints in (5.2)–(5.5) and the orthogonality conditions (5.11) and (5.12) are fulfilled. This defines a so-called *linear complementarity problem* or LCP (see Cottle, Pang, and Stone 1992).

Theorem 5.2. The equilibria of a general two-person game in extensive form with perfect recall are the solutions of a linear complementarity problem whose size, in sparse representation, is linear in the size of the game tree.

By using the sequence form, the LCP is of small size. A very similar LCP can be formulated for a game in normal form, but considered for a game in extensive form it may have exponential size and the matrices are full and not sparse. For bimatrix games, the algorithm by Lemke and Howson (1964) finds at least one LCP solution, that is, one equilibrium. Unfortunately, the LCP for the sequence form is not suitable for a direct application of the Lemke-Howson algorithm. However, one can use the more general algorithm by Lemke (1965) instead. This is shown in detail by Koller, Megiddo, and von Stengel (1995).

Finally, we note an obvious reduction of the constraint and payoff matrices which reduces the size of the LP in (5.6) and of the mentioned LCP even further. It is possible to consider only those sequences of a player that actually lead to a leaf of the game tree. Other sequences σ_u, like the empty sequence $\emptyset$ for player 1 in Fig. 2.1, have only zero entries in both payoff matrices. Therefore, the variable $r_i(\sigma_u)$ does not appear in the payoff term. In the constraints (5.1), this variable can be eliminated as follows: There is one equation where $r_i(\sigma_u)$ has coefficient 1, and one or more equations where it is the first variable in (3.4) with a negative sign. For each of the latter equations, take the corresponding sum in (3.4) and substitute it for $r_i(\sigma_u)$ in the first equation. The resulting equations replace the old ones. This eliminates one variable and one equation. If $\sigma_u = \emptyset$, the discarded equation is (3.3) and the right hand side must also be observed in these replacements. All variables $r_i(\sigma_u)$ corresponding to non-terminal sequences σ_u can be eliminated in this way. In the example of Fig. 2.1, the constraint matrix E and the right hand side e in (5.1) are thus reduced to

$$E = \begin{pmatrix} 1 & 1 & & \\ & & 1 & 1 \end{pmatrix} \qquad \text{and} \qquad e = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Thereby, the component $x_\emptyset$ of the vector x is omitted and the payoff matrix has to be sized accordingly.

By this method, the total number of variables is reduced by less than a half, since for each choice preceding an information set there are at least two other choices, at that set, further down the tree. Therefore, the reduction is not significant, in particular since (5.1) is always very easy to solve. The reduction has another disadvantage: The constraints (3.4) are quite canonical since they correspond to the information sets of the players. As described above, they also correspond to dual LP variables, the components of p and q. As we will show in the next section, these can be interpreted as 'payoff contributions' of optimal choices at the respective information sets. In certain applications, this interpretation may be interesting, but it is destroyed by the indicated reduction.

6. Best Responses for Any Number of Players

In the previous section, we described the linear program for computing a best response of one player against a fixed strategy of the other player. The same can be done for any number of players, where all but one of them play with a fixed strategy. We will interpret the solutions of this LP with the help of the dual variables and the complementary slackness condition. This characterizes a behavior strategy 'locally' as a best response, in the sense that it chooses only moves yielding a maximal payoff 'contribution' with positive probability. Furthermore, we will describe a nonlinear optimization problem whose solutions are the equilibria of the N-person game.

For illustration, we consider first a two-person game in normal form, that is, a bimatrix game. Let it have payoff matrices A and B, and let x and y denote mixed strategies of player 1 and 2, respectively. Then, their expected payoffs are x^TAy and x^TBy as above. Assume that y is fixed. A best response x of player 1 is a solution to the LP (5.4) where E is a matrix consisting of the single row $(1\ 1 \cdots 1)$ and e is the scalar 1. In fact, this constraint matrix for the bimatrix game can be regarded as a special case of our approach for the sequence form: To see this, convert the bimatrix game to an extensive game in the usual way, with one information set for each player where his choices are his strategies in the given normal form. Then, the sequence form is equal to the normal form except for the empty sequence $\emptyset$, which, however, is not defined by a leaf of the game tree and disappears if the sequence form is reduced as described at the end of Section 5.

For the bimatrix game, the dual LP (5.5) has a single scalar variable p, whose optimum is the maximum of the components $(Ay)_j$ of the expected payoff vector Ay. In other words, p is the best possible payoff for a pure strategy j of player 1. The complementary slackness condition (5.11) says that x is a best response to y if and only if for all j,

$$x_j > 0 \quad \Longrightarrow \quad (E^T p)_j = (Ay)_j\,, \tag{6.1}$$

where $(E^T p)_j$ is here simply p. Condition (6.1) states that only best responses j are chosen with positive probability x_j, which is the familiar criterion of a best response due to Nash (1951, p. 287).

The same can be done with the sequence form. We consider first, in a technical way, the example of Fig. 2.1, and will explain the general case later. Let y be a fixed realization plan for player 2, say $y = (1, {}^2\!/_3, {}^1\!/_3)^T$ in correspondence to the sequences $\emptyset, c, d$ (this is not the optimal realization plan for player 2). The LP (5.5) has the dual vector $p = (p_0, p_v, p_{v'})$ of variables, where p_0 is the first component and the other components correspond to the information sets of player 1. The constraints $E^T p \geq Ay$ have the form

$$\begin{pmatrix} 1 & -1 & -1 \\ & 1 & \\ & 1 & \\ & & 1 \\ & & 1 \end{pmatrix} p \;\geq\; \begin{pmatrix} & & \\ 0 & & \\ & 1 & -1 \\ & -2 & 4 \\ 1 & & \end{pmatrix} y \;=\; \begin{pmatrix} 0 \\ 0 \\ {}^1\!/_3 \\ 0 \\ 1 \end{pmatrix}. \tag{6.2}$$

The objective function $e^T p$, that is, p_0, is minimized by $p_{v'} = 1$, $p_v = {}^1\!/_3$ and $p_0 = {}^4\!/_3$. The expected payoff that player 1 receives is p_0. Only the first, third, and fifth inequalities in (6.2) are tight, that is, hold with equality, the others have positive slacks. Therefore, by the complementary slackness condition (5.11), analogous to (6.1), only $x_\emptyset$, x_r, and x_R can be positive, so that the unique best response of player 1 is $x = (1, 0, 1, 0, 1)^T$ in correspondence to her sequences $\emptyset, l, r, L, R$. For the optimal realization plan $y = (1, {}^1\!/_2, {}^1\!/_2)^T$ of player 2, the best response of player 1 is not unique since all inequalities in (6.2) are tight. As in the normal form, we see a connection between complementary slackness and best responses. We make this explicit in the following discussion.

We consider now a general game with any number N of players, and assume that all players play according to certain realization plans except for one player, say player 1. As before, let her realization plan r_1 be denoted by x, and now let y denote the given realization plans $r_2, \ldots, r_N$ of all remaining players. We are interested in finding and characterizing the best responses x to y. Analogous to the components of Ay for the two-person game, we denote the *payoff contribution* of the sequence s_1 by

$$G_1(s_1, y) = \sum_{s_0, s_2, \ldots, s_N} g_1(s_0, s_1, s_2, \ldots, s_N)\ r_0(s_0) r_2(s_2) \cdots r_N(s_N), \tag{6.3}$$

where the sum is taken over all $s_i \in S_i$ for $i = 0, 2, \ldots, N$. If we denote $r_1(s_1)$ by $x(s_1)$, then the expected payoff to player 1 in (3.7) has the form

$$G_1(x, y) = \sum_{s_1 \in S_1} x(s_1)\, G_1(s_1, y)\,,$$

which corresponds to the term $x^T(Ay)$ for the two-person game. As before, this is a linear expression in the variables $x(s_1)$ for $s_1 \in S_1$. Maximizing it subject to $Ex = e$ as in (5.4) determines a best response x to y. For the dual LP (5.5), the vector p of dual variables has the components p_0, corresponding to the first equation (3.3) of $Ex = e$, and p_u for $u \in U_1$. Then (5.5) says: Minimize p_0 subject to

$$p_0 - \sum_{v \in U_1,\ \sigma_v = \emptyset} p_v \geq G_1(\emptyset, y) \tag{6.4}$$

and

$$p_u - \sum_{v \in U_1,\ \sigma_v = \sigma_u c} p_v \geq G_1(\sigma_u c, y) \qquad \text{for } u \in U_1, \quad c \in C_u. \tag{6.5}$$

An example of these constraints is (6.2). The entries of the constraint matrix E appear here in E^T as follows: Each inequality corresponds to a sequence s_1 in S_1. The coefficient 1 of $r_1(s_1)$ in the primal constraints (3.3) and (3.4) is the coefficient of p_0 in (6.4) for $s_1 = \emptyset$, and of p_u in (6.5) for a nonempty sequence $s_1 = \sigma_u c$. For each information set v in U_1, there is an additional coefficient -1 in (3.4). It is the coefficient of p_v in (6.4) or (6.5) for the sequence $s_1 = \sigma_v$ that leads to v, where parallel information sets v appear in a single inequality.

With this notation, the complementary slackness condition (5.11) that characterizes x and p as optimal solutions to (5.4) and (5.5) has the following form, analogous to (6.1):

$$p_0 = G_1(\emptyset, y) + \sum_{v \in U_1,\ \sigma_v = \emptyset} p_v \tag{6.6}$$

because $x(\emptyset)$ is always positive, and

$$x(\sigma_u c) > 0 \quad \Longrightarrow \quad p_u = G_1(\sigma_u c, y) + \sum_{v \in U_1,\ \sigma_v = \sigma_u c} p_v \tag{6.7}$$

for $u \in U_1,\ c \in C_u$. When these conditions hold, the two objective functions are equal, and $G_1(x, y)$ and p_0 represent the maximal expected payoff to player 1.

For the realization plan x and its corresponding behavior strategy β_1, (6.7) can be interpreted as a 'best response criterion' in terms of the player's moves. As (6.2) illustrates, the constraint matrix E has a simple structure resulting from the game tree. Therefore, a best response x to y and an optimal dual vector p can be found directly by backward induction or 'dynamic programming', without using an LP algorithm. In this inductive procedure, one starts with the information sets u that are closest to the leaves and constructs an optimal behavior at u as well as optimal dual variables p_u. For these information sets u, the sum over the sets v in (6.5) is empty, and p_u can be defined as the maximum of the payoff contributions $G_1(\sigma_u c, y)$ for the choices c at u. (In the example (6.2), these dual variables are $p_{v'} = 1$ and $p_v = {}^1\!/_3$. They can be interpreted in Fig. 2.1: Player 2, using y, makes the choices c and d with probabilities ${}^2\!/_3$ and ${}^1\!/_3$, respectively. The resulting expected payoffs have to multiplied by the chance probabilities to determine the payoff contributions for player 1. For her information set v', these are 0 and 1 for her sequences L and R, shown in (6.2), so her optimal payoff contribution is 1 with the sequence R. Similarly, at her information set v it is ${}^1\!/_3$ with her sequence r.) To make sure that (6.7) holds, the behavior strategy β_1 should assign positive probabilities $\beta_1(c)$, which can otherwise be arbitrary, only to the 'optimal' choices c at u where the payoff contribution is actually maximal. (In the above example, this implies the deterministic behavior of player 1 at v' and v.) If it turns out later that u is irrelevant, the equation $x(\sigma_u c) = x(\sigma_u) \cdot \beta_1(c)$ implies that the behavior at u can be arbitrary, but for $x(\sigma_u) > 0$ it is necessary for (6.7) that only optimal moves are made with positive probability.

The dual variable p_u and optimal moves at u are defined inductively at information sets u higher up in the tree, where the dual variables p_v with $\sigma_v = \sigma_u c$ for the choices c at u and the behavior at v are already known. Thereby, p_u can be interpreted as the maximal payoff contribution of a move c at u, regarded as the collective contribution of the sequence $\sigma_u c$ and of all longer sequences as expressed by the sum in (6.7). This payoff is also achieved if the choices further down the tree are made optimally. Eventually, this defines a behavior strategy β_1 that is a best response to y and the realization plan $x = r_1$ of β_1 according to (3.1). Finally, p_0 as defined by (6.6) is the expected payoff $G_1(x, y)$ to the player at the root of the tree (in the above example, $p_0 = 0 + p_v + p_{v'} = {}^4\!/_3$).

This best response criterion for behavior strategies says that at each information set, only 'locally optimal' moves, looking down the tree, should be made with positive probability, which is in some sense analogous to (6.1) for the normal form. As there, any best response can be deterministic. However, a behavior strategy is not necessarily a best response if it is 'locally optimal' in terms of the overall payoff, that is, if the payoff cannot be improved by changing the behavior at a single information set. An example (due to Robert Aumann, personal communication) is Fig. 4.1 where the pure strategy (r, R) is a behavior strategy that has to be changed at both information sets to improve the payoff. Here, the choice R is optimal only in terms of the global payoff since it is made at an irrelevant information set u, but it is not optimal in the inductive construction, and a corresponding dual variable $p_u = 0$ would not be feasible since (6.5) would be violated for the sequence $\sigma_u c = l\,L$.

A best response behavior strategy is determined by the 'bottom up' induction just described. Nevertheless, it need not be part of a *subgame perfect* equilibrium. If an information set u is not reached due to the behavior y of the *other* players, then all choices $c \in C_u$ produce the equally 'optimal' payoff contribution $G_1(\sigma_u c, y) = 0$ in (6.5). These arbitrary moves need not be optimal in a subgame, as it is required by subgame perfectness (see Selten 1975, p. 33).

To conclude this section, we describe a nonlinear optimization problem whose solutions are the equilibria of an N-person game. In this problem, the components of all realization plans $r_1, \ldots, r_N$ are treated as variables. The dual constraints (6.4) and (6.5) are regarded for all players, and the goal is to close the 'dual gap' between dual and primal objective function for all players simultaneously. Notationally, we replace the subscript 1 indicating player 1 in (6.3), (6.4) and (6.5) by i for $1 \le i \le N$. Correspondingly, y represents $(r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_N)$, and (6.3) has the form

$$G_i(s_i, y) = \sum_{s_j \in S_j,\, j \neq i} g_i(s_0, s_1, \ldots, s_N) \prod_{j \neq i,\, 0 \le j \le N} r_j(s_j). \tag{6.8}$$

For each player i, we continue to denote the $1 + |U_i|$ components of the dual vector by p_u (indexed by the respective information sets $u \in U_i$), except for the first component p_0 that shall be called d_i, also representing the value of the dual objective function. The value of the primal objective function is $G_i(r)$ as in (3.7). Generalizing the inequalities in (5.3) and (5.5), we obtain the following constraints from (6.4) and (6.5), with the subscript 1 replaced by $i = 1, \ldots, N$:

$$-\,G_i(\emptyset, y) \,+\, d_i - \sum_{v \in U_i,\ \sigma_v = \emptyset} p_v \ge 0 \tag{6.9}$$

and

$$-\,G_i(\sigma_u c, y) \,+\, p_u - \sum_{v \in U_i,\ \sigma_v = \sigma_u c} p_v \ge 0 \qquad \text{for } u \in U_i, \quad c \in C_u. \tag{6.10}$$

Note that for $N > 2$ these are nonlinear constraints since the expression in (6.8) is nonlinear. With this notation, we state the following result.

Theorem 6.1. Consider an extensive N-person game with perfect recall and its sequence form. Let r_i be the realization plan of a behavior strategy β_i of player i, and let $r = (r_1, \ldots, r_N)$. Then $(\beta_1, \ldots, \beta_N)$ is an equilibrium if and only if the reals $r_i(s_i)$ for $s_i \in S_i$ and suitable reals d_i and p_u for $u \in U_i$ solve the following optimization problem:

$$\text{minimize} \quad \sum_{i=1}^{N} \Big(d_i - G_i(r) \Big) \tag{6.11}$$

subject to (3.3), (3.4), (3.5), (6.9) and (6.10), for $1 \le i \le N$.

Proof. In equilibrium, each behavior strategy β_i is a best response to the others. An equivalent condition is that for each i, $1 \le i \le N$, and fixed y, the primal LP of maximizing the expected payoff $G_i(r)$ to player i subject to the constraints (3.3), (3.4) and (3.5), and its dual of minimizing d_i subject to (6.9) and (6.10), are solved optimally, with $d_i = G_i(r)$. For any feasible solution, $d_i \ge G_i(r)$ by weak duality, so the expression in (6.11) is nonnegative, and its minimum is attained at zero since an equilibrium always exists. □

This optimization problem generalizes the LP and LCP for a two-person game described in the previous section. With the notation used there, the objective function in (6.11) is $(e^T p - x^T Ay) + (q^T f - x^T By)$, which is linear for a zero-sum game with $A + B = 0$. Then, the problem is a self-dual LP with the constraints of both (5.6) and (5.7), and equivalent to either LP. For a general two-person game, the optimization problem is equivalent to the LCP mentioned in Section 5 since (6.11) can also be written as a minimization of $x^T(-Ay + E^T p) \ + \ y^T(-B^T x + F^T q)$, which has its minimum at zero as required by (5.11) and (5.12).

Algorithms to solve the multilinear optimization problem in Theorem 6.1 for more than two players have been described by Rosenmüller (1971) and Wilson (1971) for normal form games. Howson (1972) has solved special N-person games that are essentially equivalent to a set of simultaneously played bimatrix games. There should be no major difficulty in applying this algorithm, and other methods for solving normal form games, to the sequence form.

7. Comparison with Related Work and Conclusions

In this section, we summarize our contributions and explain their relationship to earlier work. The most closely related papers are Koller and Megiddo (1992) and Wilson (1972), which introduced some techniques that we have used. We will also discuss games without perfect recall, and conclude with perspectives for further research.

For extensive zero-sum games with perfect recall, we have shown in Theorem 5.1 that optimal strategies for both players are determined by a small LP. There are two main ideas behind this approach. First, the realization probabilities for sequences of choices are used for optimization. The expected payoff is linear in these variables, which is not true for behavior strategy probabilities. Second, we have taken the LP of finding a best reponse of one player against a fixed strategy of the other player and considered its dual. In this dual LP (5.5), the dual variables are separated from the decision variables of the other player so that one obtains linear constraints that can be used in the LP (5.6) for the entire game. In summary, we have linearized the problem of using behavior strategy probabilities by a variable transformation (introducing products over sequences as in (3.1) as new variables), and by a suitable separation of variables.

The first of these ideas is not new. Koller and Megiddo (1992) presented the first polynomial-time algorithm for solving an extensive zero-sum game with perfect recall. They described a behavior strategy by nonnegative variables that are subject to linear equations analogous to (3.4). Instead of sequences of choices, they considered nodes of the game tree for defining realization plans, but this is not essentially different (see also Koller and Megiddo 1992, Remark 3.8, p. 545). A minor point is that they overlooked the possibility $r_i(\sigma_u) = 0$ in (3.6), so the behavior strategy β_i cannot always be uniquely reconstructed from the realization plan r_i as they claimed (Prop. 3.6, p. 543).

The main difference to our approach is that Koller and Megiddo (1992, p. 546) still considered pure strategies of the opponent. Namely, if player 1, say, plays according to realization plan x, and player 2 uses the pure strategy π_2, then the expected payoff $G_1(x, \pi_2)$ to player 1 is a linear function of x. The problem of finding a max-min strategy x for player 1 can then, in our notation, be written as: Maximize λ subject to $\lambda \le G_1(x, \pi_2)$ for all pure strategies π_2 of player 2, and so that x defines a realization plan. This is an LP for computing an optimal realization plan for player 1, but with a generally exponential number of inequalities. However, the *separation problem* for this LP can be solved in polynomial time. Given a vector x and a real number λ, the separation problem is to either verify that all constraints of the LP are satisfied, or, if not, to find a violated constraint. Given a realization plan x, this is solved by computing a best response π_2 of player 2, which can be done fast. Using the ellipsoid algorithm for linear programming, it suffices to evaluate only polynomially many constraints, so that the LP can be solved in polynomial time. (The ellipsoid algorithm is not very practical. For the dual LP, however, one could use in the same way the simplex algorithm with 'column generation'.)

In contrast to this approach by Koller and Megiddo, we obtained an LP with a linear number of variables and constraints by employing the indicated duality. Our payoff matrix is of small size since we use sequences symmetrically for both players. This was partly motivated by a 'strategic' view of sequences as a replacement for strategies. For that reason, we have also considered choices and not nodes of the game tree for defining realization plans.

Another property of the sequence form, for any number of players, is that it allows to verify quickly whether an N-tuple $\beta_1, \ldots, \beta_N$ of behavior strategies represents an equilibrium: The realization plans of these behavior strategies are determined by (3.1), and the conditions (6.6) and (6.7) show that a realization plan (for player 1, similarly for the other players) is a best response to the others. If the normal form is used for verifying the equilibrium property according to (6.1), then it may be necessary to evaluate exponentially many pure strategies.

The fast verification of an equilibrium in behavior strategies has also been known earlier (this has been pointed out to the author by Daphne Koller). Given the behavior strategies of the other players, a best response pure strategy can be found by backward induction as described by Wilson (1972, p. 455): Using the condition of perfect recall, the game is converted to an equivalent one-person game with singleton information sets. Then, a best response is found easily. If the given behavior strategy yields the same payoff, it is also a best response. All computations take linear time in the size of the game tree. The best response that has been found serves as a dynamically generated pivot column for the Lemke-Howson algorithm (Wilson 1972, p. 458). A backward induction method that is closer to ours was described by Koller and Megiddo (1992, p. 547). There, the generated best response solves the separation problem for the LP mentioned above.

Koller and Megiddo (1992) also studied extensive games without perfect recall. We will discuss how the sequence form can be applied to this case. A type of game that is still tractable is a game where each player has *perfect memory.* By definition, this means that if a node b of an information set v of the player is preceded by a choice at an information set u of the same player, then each node in v is preceded by *some* choice at u (for perfect recall, which is stronger, it would have to be the same choice). That is, a player with perfect memory may forget earlier moves but not earlier knowledge.

Koller and Megiddo (1992, p. 550) showed that in an extensive zero-sum game, one can compute in polynomial time a max-min behavior strategy for player 1, say, who has perfect recall, if her opponent has only perfect memory. They considered the *complete inflation* of the game obtained by partitioning the information sets of player 2 such that he can distinguish his earlier moves (see Dalkey 1953, p. 226, Okada 1985, p. 90). In the modified game, player 2 also has perfect recall, and his best response π_2 to a realization plan x can be computed as before. Furthermore, his best response can be reinterpreted for the original game. Essentially, this is possible since the game has the same reduced normal form as its complete inflation (Dalkey 1953, p. 228).

Using the sequence form, we obtain the following stronger result: For a two-person game in extensive form with perfect memory, Theorems 5.1 and 5.2 hold for zero-sum payoffs and general payoffs, respectively, provided a player without perfect recall may use *mixed* strategies. This is easily seen by considering the complete inflation of the game. It has perfect recall, so the sequence form can be applied and the theorems hold. Equilibrium realization plans can be translated to mixed strategies in the reduced normal form of the game (see Section 4 above). Since the original game has the same reduced normal form, they are also equilibrium strategies there. However, they may no longer be behavior strategies for a player without perfect recall, since the probability for a move at an information set that is partitioned in the complete inflation may depend on an earlier move.

Otherwise, it seems that the sequence form cannot be applied to games without perfect recall. For a zero-sum game, Koller and Megiddo (1992, p. 534) have shown that it is NP-hard to find an optimal mixed strategy for a player without perfect recall. Furthermore, a max-min *behavior* strategy may involve irrational numbers, even if the player has perfect memory (p. 537), so it cannot be the solution of an LP. If a player has no perfect memory as represented by his information sets and is assumed to 'forget' during play, then a proper definition of the sequence form of the game and its intended 'strategic' interpretation is also difficult.

As a topic for further research, the sequence form may be of conceptual interest because of its correspondence to the game tree. The constraints in the sequence form are closely related to backward induction. That notion is used to distinguish certain equilibria as perfect or subgame perfect (see Selten 1975), and in many other definitions of 'stability' of an equilibrium (see Kohlberg and Mertens 1986, van Damme 1987). So far, the equilibria that can be computed with the sequence form are arbitrary and need not induce an equilibrium in every subgame, in which case they are not subgame perfect. On the other hand, this should be easy to accomplish, since equilibria of subgames can be computed with the presented algorithms.

Acknowledgements

The author is indebted to Robert Aumann, Morton Canty, Sergiu Hart, Harold Kuhn, Nimrod Megiddo, Bezalel Peleg, Robert Wilson, Shmuel Zamir, and a referee for stimulating discussions and helpful comments, and to Rudolf Avenhaus for continuing support. Particular thanks are due to Daphne Koller for detailed suggestions that led to a major reorganization of the text. Financial support was provided by the Volkswagen Foundation and by the German-Israeli Foundation for Scientific Research and Development (G.I.F.).

References

V. Chvátal (1983), Linear Programming. Freeman, New York.

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

N. Dalkey (1953), Equivalence of information patterns and essentially determinate games. In: Contributions to the Theory of Games II, eds. H. W. Kuhn and A. W. Tucker, Annals of Mathematics Studies 28, Princeton Univ. Press, Princeton, 217–243.

G. B. Dantzig (1963), Linear Programming and Extensions. Princeton University Press, Princeton.

D. Gale (1960), The Theory of Linear Economic Models. McGraw-Hill, New York.

J. T. Howson, Jr. (1972), Equilibria of polymatrix games. Management Science 18, 312–318.

E. Kohlberg and J.-F. Mertens (1986), On the strategic stability of equilibria. Econometrica 54, 1003–1037.

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 (1995), Finding mixed strategies with small supports in extensive games. International Journal of Game Theory, to appear.

D. Koller, N. Megiddo, and B. von Stengel (1994), Fast algorithms for finding randomized strategies in game trees. Proceedings of the 26th ACM Symposium on Theory of Computing, 750–759.

D. Koller, N. Megiddo, and B. von Stengel (1995), Efficient computation of equilibria for extensive two-person games. Games and Economic Behavior, this issue.

H. W. Kuhn (1950), Extensive games. Proceedings of the National Academy of Sciences of the U.S.A. 36, 570–576.

H. W. Kuhn (1953), Extensive games and the problem of information. In: Contributions to the Theory of Games II, eds. H. W. Kuhn and A. W. Tucker, Annals of Mathematics Studies 28, Princeton Univ. Press, Princeton, 193–216.

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

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

W. F. Lucas (1972), An overview of the mathematical theory of games. Management Science 15, Appendix P, 3–19.

J. F. Nash (1951), Non-cooperative games. Annals of Mathematics 54, 286–295.

A. Okada (1987), Complete inflation and perfect recall in extensive games. International Journal of Game Theory 16, 85–91.

J. Rosenmüller (1971), On a generalization of the Lemke-Howson algorithm to noncooperative N-person games. SIAM Journal on Applied Mathematics 21, 73–79.

R. Selten (1975), Reexamination of the perfectness concept for equilibrium points in extensive games. International Journal of Game Theory 4, 22–55.

J. Swinkels (1989), Subgames and the reduced normal form. Econometric Research Program, Princeton University, Research Memorandum No. 344.

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

B. von Stengel (1993), LP representation and efficient computation of behavior strategies. Technical Report S-9301, revised version submitted in 1994 as Habilitation thesis, Faculty for Computer Science, University of the Federal Armed Forces at Munich.

R. Wilson (1971), Computing equilibria of N-person games. SIAM Journal on Applied Mathematics 21, 80–87.

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