# 3

# Introduction to Noncooperative Game Theory: Games in Normal Form

Game theory is the mathematical study of interaction among independent, self-interested agents. It has been applied to disciplines as diverse as economics (historically, its main area of application), political science, biology, psychology, linguistics—and computer science. In this chapter we will concentrate on what has become the dominant branch of game theory, called *noncooperative* game theory, and specifically on normal-form games, a canonical representation in this discipline.

As an aside, the name "noncooperative game theory" could be misleading, since it may suggest that the theory applies exclusively to situations in which the interests of different agents conflict. This is not the case, although it is fair to say that the theory is most interesting in such situations. By the same token, in Chapter 12 we will see that *coalitional game theory* (also known as *cooperative game theory*) does not apply only in situations in which the interests of the agents align with each other. The essential difference between the two branches is that in noncooperative game theory the basic modeling unit is the individual (including his beliefs, preferences, and possible actions) while in coalitional game theory the basic modeling unit is the group. We will return to that later in Chapter 12, but for now let us proceed with the individualistic approach.

coalitional game theory

## 3.1 Self-interested agents

What does it mean to say that agents are self-interested? It does not necessarily mean that they want to cause harm to each other, or even that they care only about themselves. Instead, it means that each agent has his own description of which states of the world he likes—which can include good things happening to other agents—and that he acts in an attempt to bring about these states of the world. In this section we will consider how to model such interests.

utility theory

The dominant approach to modeling an agent's interests is *utility theory*. This theoretical approach aims to quantify an agent's degree of preference across a set of available alternatives. The theory also aims to understand how these preferences change when an agent faces uncertainty about which alternative he will receive. When we refer to an agent's *utility function*, as we will do throughout much of this book, we will be making an implicit assumption that the agent has

utility function

desires about how to act that are consistent with utility-theoretic assumptions. Thus, before we discuss game theory (and thus interactions between *multiple* utility-theoretic agents), we should examine some key properties of utility functions and explain why they are believed to form a solid basis for a theory of preference and rational action.

A utility function is a mapping from states of the world to real numbers. These numbers are interpreted as measures of an agent's level of happiness in the given states. When the agent is uncertain about which state of the world he faces, his utility is defined as the expected value of his utility function with respect to the appropriate probability distribution over states.

### 3.1.1 *Example: friends and enemies*

We begin with a simple example of how utility functions can be used as a basis for making decisions. Consider an agent Alice, who has three options: going to the club ($c$), going to a movie ($m$), or watching a video at home ($h$). If she is on her own, Alice has a utility of 100 for $c$, 50 for $m$, and 50 for $h$. However, Alice is also interested in the activities of two other agents, Bob and Carol, who frequent both the club and the movie theater. Bob is Alice's nemesis; he is downright painful to be around. If Alice runs into Bob at the movies, she can try to ignore him and only suffers a disutility of 40; however, if she sees him at the club he will pester her endlessly, yielding her a disutility of 90. Unfortunately, Bob prefers the club: he is there 60% of the time, spending the rest of his time at the movie theater. Carol, on the other hand, is Alice's friend. She makes everything more fun. Specifically, Carol increases Alice's utility for either activity by a factor of 1.5 (after taking into account the possible disutility of running into Bob). Carol can be found at the club 25% of the time, and the movie theater 75% of the time.

It will be easier to determine Alice's best course of action if we list Alice's utility for each possible state of the world. There are 12 outcomes that can occur: Bob and Carol can each be in either the club or the movie theater, and Alice can be in the club, the movie theater, or at home. Alice has a baseline level of utility for each of her three actions, and this baseline is adjusted if either Bob, Carol, or both are present. Following the description of our example, we see that Alice's utility is always 50 when she stays home, and for her other two activities it is given by Figure 3.1.

So how should Alice choose among her three activities? To answer this question we need to combine her utility function with her knowledge of Bob and Carol's randomized entertainment habits. Alice's expected utility for going to the club can be calculated as $0.25(0.6 \cdot 15 + 0.4 \cdot 150) + 0.75(0.6 \cdot 10 + 0.4 \cdot 100) = 51.75$. In the same way, we can calculate her expected utility for going to the movies as $0.25(0.6 \cdot 50 + 0.4 \cdot 10) + 0.75(0.6(75) + 0.4(15)) = 46.75$. Of course, Alice gets an expected utility of 50 for staying home. Thus, Alice prefers to go to the club (even though Bob is often there and Carol rarely is) and prefers staying home to going to the movies (even though Bob is usually not at the movies and Carol almost always is).

**A = c**

| | $B = c$ | $B = m$ |
|---|---|---|
| $C = c$ | 15 | 150 |
| $C = m$ | 10 | 100 |

**A = m**

| | $B = c$ | $B = m$ |
|---|---|---|
| $C = c$ | 50 | 10 |
| $C = m$ | 75 | 15 |

**Figure 3.1** Alice's utility for the actions $c$ and $m$.

### 3.1.2 *Preferences and utility*

Because the idea of utility is so pervasive, it may be hard to see why anyone would argue with the claim that it provides a sensible formal model for reasoning about an agent's happiness in different situations. However, when considered more carefully this claim turns out to be substantive, and hence requires justification. For example, why should a single-dimensional function be enough to explain preferences over an arbitrarily complicated set of alternatives (rather than, say, a function that maps to a point in a three-dimensional space, or to a point in a space whose dimensionality depends on the number of alternatives being considered)? And why should an agent's response to uncertainty be captured purely by the expected value of his utility function, rather than also depending on other properties of the distribution such as its standard deviation or number of modes?

preferences

Utility theorists respond to such questions by showing that the idea of utility can be grounded in a more basic concept of *preferences*. The most influential such theory is due to von Neumann and Morgenstern, and thus the utility functions are sometimes called von Neumann–Morgenstern utility functions to distinguish them from other varieties. We present that theory here.

Let $O$ denote a finite set of outcomes. For any pair $o_1, o_2 \in O$, let $o_1 \succeq o_2$ denote the proposition that the agent weakly prefers $o_1$ to $o_2$. Let $o_1 \sim o_2$ denote the proposition that the agent is indifferent between $o_1$ and $o_2$. Finally, by $o_1 \succ o_2$, denote the proposition that the agent strictly prefers $o_1$ to $o_2$. Note that while the second two relations are notationally convenient, the first relation $\succeq$ is the only one we actually need. This is because we can define $o_1 \succ o_2$ as "$o_1 \succeq o_2$ and not $o_2 \succeq o_1$," and $o_1 \sim o_2$ as "$o_1 \succeq o_2$ and $o_2 \succeq o_1$."

lottery

We need a way to talk about how preferences interact with uncertainty about which outcome will be selected. In utility theory this is achieved through the concept of *lotteries*. A lottery is the random selection of one of a set of outcomes according to specified probabilities. Formally, a lottery is a probability distribution over outcomes written $[p_1 : o_1, \ldots, p_k : o_k]$, where each $o_i \in O$, each $p_i \geq 0$ and $\sum_{i=1}^{k} p_i = 1$. We will extend the $\succeq$ relation to apply to lotteries as well as to the elements of $O$, effectively considering lotteries over outcomes to be outcomes themselves.

We are now able to begin stating the axioms of utility theory. These are constraints on the $\succeq$ relation which, we will argue, make it consistent with our ideas of how preferences should behave.

**Axiom 3.1.1 (Completeness)** $\forall o_1, o_2,\ o_1 \succ o_2$ *or* $o_2 \succ o_1$ *or* $o_1 \sim o_2$.

The completeness axiom states that the $\succeq$ relation induces an ordering over the outcomes, allowing ties. For every pair of outcomes, either the agent prefers one to the other or he is indifferent between them.

**Axiom 3.1.2 (Transitivity)** *If* $o_1 \succeq o_2$ *and* $o_2 \succeq o_3$*, then* $o_1 \succeq o_3$.

There is good reason to feel that every agent should have transitive preferences. If an agent's preferences were nontransitive, then there would exist some triple of outcomes $o_1$, $o_2$, and $o_3$ for which $o_1 \succeq o_2$, $o_2 \succeq o_3$, and $o_3 \succ o_1$. We can show that such an agent would be willing to engage in behavior that is hard to call rational. Consider a world in which $o_1$, $o_2$, and $o_3$ correspond to owning three different items, and an agent who currently owns the item $o_3$. Since $o_2 \succeq o_3$, there must be some nonnegative amount of money that the agent would be willing to pay in order to exchange $o_3$ for $o_2$. (If $o_2 \succ o_3$ then this amount would be strictly positive; if $o_2 \sim o_3$, then it would be zero.) Similarly, the agent would pay a nonnegative amount of money to exchange $o_2$ for $o_1$. However, from non-transitivity ($o_3 \succ o_1$) the agent would *also* pay a strictly positive amount of money to exchange $o_1$ for $o_3$. The agent would thus be willing to pay a strictly positive sum to exchange $o_3$ for $o_3$ in three steps. Such an agent could quickly be separated from *any* amount of money, which is why such a scheme is known as
money pump a *money pump*.

**Axiom 3.1.3 (Substitutability)** *If* $o_1 \sim o_2$*, then for all sequences of one or more outcomes* $o_3, \ldots, o_k$ *and sets of probabilities* $p, p_3, \ldots, p_k$ *for which* $p + \sum_{i=3}^{k} p_i = 1$,

$$[p : o_1, p_3 : o_3, \ldots, p_k : o_k] \sim [p : o_2, p_3 : o_3, \ldots, p_k : o_k].$$

Let $P_\ell(o_i)$ denote the probability that outcome $o_i$ is selected by lottery $\ell$. For example, if $\ell = [0.3 : o_1; 0.7 : [0.8 : o_2; 0.2 : o_1]]$, then $P_\ell(o_1) = 0.44$ and $P_\ell(o_3) = 0$.

**Axiom 3.1.4 (Decomposability)** *If* $\forall o_i \in O,\ P_{\ell_1}(o_i) = P_{\ell_2}(o_i)$ *then* $\ell_1 \sim \ell_2$.

These axioms describe the way preferences change when lotteries are introduced. Substitutability states that if an agent is indifferent between two outcomes, he is also indifferent between two lotteries that differ only in which of these outcomes is offered. Decomposability states that an agent is always indifferent between lotteries that induce the same probabilities over outcomes, no matter whether these probabilities are expressed through a single lottery or nested in a lottery over lotteries. For example, $[p : o_1, 1 - p : [q : o_2, 1 - q : o_3]] \sim [p : o_1, (1 - p)q : o_2, (1 - p)(1 - q) : o_3]$. Decomposability is sometimes called the "no fun in gambling" axiom because it implies that, all else being equal, the number of times an agent "rolls dice" has no affect on his preferences.

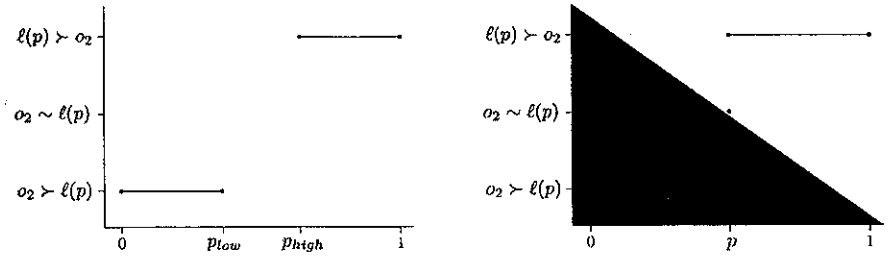


**Figure 3.2** Relationship between $o_2$ and $\ell(p)$.

**Axiom 3.1.5 (Monotonicity)** *If* $o_1 \succ o_2$ *and* $p > q$ *then* $[p : o_1, 1 - p : o_2] \succ [q : o_1, 1 - q : o_2]$.

The monotonicity axiom says that agents prefer more of a good thing. When an agent prefers $o_1$ to $o_2$ and considers two lotteries over these outcomes, he prefers the lottery that assigns the larger probability to $o_1$. This property is called monotonicity because it does not depend on the numerical values of the probabilities—the more weight $o_1$ receives, the happier the agent will be.

**Lemma 3.1.6** *If a preference relation* $\succeq$ *satisfies the axioms completeness, transitivity, decomposability, and monotonicity, and if* $o_1 \succ o_2$ *and* $o_2 \succ o_3$*, then there exists some probability* $p$ *such that for all* $p' < p$, $o_2 \succ [p' : o_1; (1 - p') : o_3]$*, and for all* $p'' > p$, $[p'' : o_1; (1 - p'') : o_3] \succ o_2$.

**Proof.** Denote the lottery $[p : o_1; (1 - p) : o_3]$ as $\ell(p)$. Consider some $p_{low}$ for which $o_2 \succ \ell(p_{low})$. Such a $p_{low}$ must exist since $o_2 \succ o_3$; for example, by decomposability $p_{low} = 0$ satisfies this condition. By monotonicity, $\ell(p_{low}) \succ \ell(p')$ for any $0 \leq p' < p_{low}$, and so by transitivity $\forall p' \leq p_{low}$, $o_2 \succ \ell(p')$. Consider some $p_{high}$ for which $\ell(p_{high}) \succ o_2$. By monotonicity, $\ell(p') \succ \ell(p_{high})$ for any $1 \geq p' > p_{high}$, and so by transitivity $\forall p' \geq p_{high}$, $\ell(p') \succ o_2$. We thus know the relationship between $\ell(p)$ and $o_2$ for all values of $p$ except those on the interval $(p_{low}, p_{high})$. This is illustrated in Figure 3.2 (left).

Consider $p^* = (p_{low} + p_{high})/2$, the midpoint of our interval. By completeness, $o_2 \succ \ell(p^*)$ or $\ell(p^*) \succ o_2$ or $o_2 \sim \ell(p^*)$. First consider the case $o_2 \sim \ell(p^*)$. It cannot be that there is also another point $p' \neq p^*$ for which $o_2 \sim \ell(p')$: this would entail $\ell(p^*) \sim \ell(p')$ by transitivity, and since $o_1 \succ o_3$, this would violate monotonicity. For all $p' \neq p^*$, then, it must be that either $o_2 \succ \ell(p')$ or $\ell(p') \succ o_2$. By the arguments earlier, if there was a point $p' > p^*$ for which $o_2 \succ \ell(p')$, then $\forall p'' < p'$, $o_2 \succ \ell(p'')$, contradicting $o_2 \sim \ell(p^*)$. Similarly there cannot be a point $p' < p^*$ for which $\ell(p') \succ o_2$. The relationship that must therefore hold between $o_2$ and $\ell(p)$ is illustrated in Figure 3.2 (right). Thus, in the case $o_2 \sim \ell(p^*)$, we have our result.

Otherwise, if $o_2 \succ \ell(p^*)$, then by the argument given earlier $o_2 \succ \ell(p')$ for all $p' \leq p^*$. Thus we can redefine $p_{low}$—the lower bound of the interval of values for which we do not know the relationship between $o_2$ and $\ell(p)$—to be $p^*$. Likewise, if $\ell(p^*) \succ o_2$ then we can redefine $p_{high} = p^*$. Either way, our interval $(p_{low}, p_{high})$ is halved. We can continue to iterate the above argument, examining the midpoint of the updated interval $(p_{low}, p_{high})$. Either we will encounter a $p^*$ for which $o_2 \sim \ell(p^*)$, or in the limit $p_{low}$ will approach some $p$ from below, and $p_{high}$ will approach that $p$ from above. ■

Something our axioms do not tell us is what preference relation holds between $o_2$ and the lottery $[p : o_1; (1-p) : o_3]$. It could be that the agent strictly prefers $o_2$ in this case, that the agent strictly prefers the lottery, or that the agent is indifferent. Our final axiom says that the third alternative—depicted in Figure 3.2 (right)—always holds.

**Axiom 3.1.7 (Continuity)** *If $o_1 \succ o_2$ and $o_2 \succ o_3$, then $\exists p \in [0, 1]$ such that $o_2 \sim [p : o_1, 1-p : o_3]$.*

utility function

If we accept Axioms 3.1.1, 3.1.2, 3.1.4, 3.1.5, and 3.1.7, it turns out that we have no choice but to accept the existence of single-dimensional *utility functions* whose expected values agents want to maximize. (And if we do *not* want to reach this conclusion, we must therefore give up at least one of the axioms.) This fact is stated as the following theorem.

**Theorem 3.1.8 (von Neumann and Morgenstern, 1944)** *If a preference relation $\succeq$ satisfies the axioms completeness, transitivity, substitutability, decomposability, monotonicity, and continuity, then there exists a function $u : O \mapsto [0, 1]$ with the properties that:*

1. *$u(o_1) \geq u(o_2)$ iff $o_1 \succeq o_2$; and*
2. *$u([p_1 : o_1, \dots, p_k : o_k]) = \sum_{i=1}^{k} p_i u(o_i)$.*

**Proof.** If the agent is indifferent among all outcomes, then for all $o_i \in O$ set $u(o_i) = 0$. In this case Part 1 follows trivially (both sides of the implication are always true), and Part 2 is immediate from decomposability.

Otherwise, there must be a set of one or more most-preferred outcomes and a disjoint set of one or more least-preferred outcomes. (There may of course be other outcomes belonging to neither set.) Label one of the most-preferred outcomes as $\overline{o}$ and one of the least-preferred outcomes as $\underline{o}$. For any outcome $o_i$, define $u(o_i)$ to be the number $p_i$ such that $o_i \sim [p_i : \overline{o}, (1-p_i) : \underline{o}]$. By continuity such a number exists; by Lemma 3.1.6 it is unique.

**Part 1:** $u(o_1) \geq u(o_2)$ iff $o_1 \succeq o_2$.

Let $\ell_1$ be the lottery such that $o_1 \sim \ell_1 = [u(o_1) : \overline{o}; 1 - u(o_1) : \underline{o}]$; similarly, let $\ell_2$ be the lottery such that $o_2 \sim \ell_2 = [u(o_2) : \overline{o}; 1 - u(o_2) : \underline{o}]$. First, we show that $u(o_1) \geq u(o_2) \Rightarrow o_1 \succeq o_2$. If $u(o_1) > u(o_2)$ then, since $\overline{o} \succ \underline{o}$ we can conclude that $\ell_1 \succ \ell_2$ by monotonicity. Thus, we have $o_1 \sim \ell_1 \succ \ell_2 \sim o_2$; by transitivity, substitutability, and decomposability, this gives $o_1 \succ o_2$. If $u(o_1) = u(o_2)$, the $\ell_1$ and $\ell_2$ are identical lotteries; thus, $o_1 \sim \ell_1 \equiv \ell_2 \sim o_2$, and transitivity gives $o_1 \sim o_2$.

Now we must show that $o_1 \succeq o_2 \Rightarrow u(o_1) \geq u(o_2)$. It suffices to prove the contrapositive of this statement, $u(o_1) \not\geq u(o_2) \Rightarrow o_1 \not\succeq o_2$, which can be rewritten as $u(o_2) > u(o_1) \Rightarrow o_2 \succ o_1$ by completeness. This statement was already proved earlier (with the labels $o_1$ and $o_2$ swapped).

**Part 2:** $u([p_1 : o_1, \ldots, p_k : o_k]) = \sum_{i=1}^{k} p_i u(o_i)$.

Let $u^* = u([p_1 : o_1, \ldots, p_k : o_k])$. From the construction of $u$ we know that $o_i \sim [u(o_i) : \overline{o}, (1 - u(o_i)) : \underline{o}]$. By substitutability, we can replace each $o_i$ in the definition of $u^*$ by the lottery $[u(o_i) : \overline{o}, (1 - u(o_i)) : \underline{o}]$, giving us $u^* = u([p_1 : [u(o_1) : \overline{o}, (1 - u(o_1)) : \underline{o}], \ldots, p_k : [u(o_k) : \overline{o}, (1 - u(o_k)) : \underline{o}]])$. This nested lottery only selects between the two outcomes $\overline{o}$ and $\underline{o}$. This means that we can use decomposability to conclude $u^* = u\left(\left[\left(\sum_{i=1}^{k} p_i u(o_i)\right) : \overline{o}, 1 - \left(\sum_{i=1}^{k} p_i u(o_i)\right) : \underline{o}\right]\right)$. By our definition of $u$, $u^* = \sum_{i=1}^{k} p_i u(o_i)$. ■

One might wonder why we do not use money to express the real-valued quantity that rational agents want to maximize, rather than inventing the new concept of utility. The reason is that while it is reasonable to assume that all agents get happier the more money they have, it is often not reasonable to assume that agents care only about the *expected values* of their bank balances. For example, consider a situation in which an agent is offered a gamble between a payoff of two million and a payoff of zero, with even odds. When the outcomes are measured in units of utility ("utils") then Theorem 3.1.8 tells us that the agent would prefer this gamble to a sure payoff of 999,999 utils. However, if the outcomes were measured in money, few of us would prefer to gamble—most people would prefer a guaranteed payment of nearly a million dollars to a double-or-nothing bet. This is not to say that utility-theoretic reasoning goes out the window when money is involved. It simply points out that utility and money are often not linearly related. This issue is discussed in more detail in Section 10.3.1.

What if we want a utility function that is not confined to the range [0, 1], such as the one we had in our friends and enemies example? Luckily, Theorem 3.1.8 does not *require* that every utility function maps to this range; it simply shows that one such utility function must exist for every set of preferences that satisfy the required axioms. Indeed, von Neumann and Morgenstern also showed that the absolute magnitudes of the utility function evaluated at different outcomes are unimportant. Instead, every positive affine transformation of a utility function yields another utility function for the same agent (in the sense that it will also satisfy both properties of Theorem 3.1.8). In other words, if $u(o)$ is a utility function for a given agent then $u'(o) = au(o) + b$ is also a utility function for the same agent, as long as $a$ and $b$ are constants and $a$ is positive.

## 3.2 Games in normal form

We have seen that under reasonable assumptions about preferences, agents will always have utility functions whose expected values they want to maximize. This suggests that acting optimally in an uncertain environment is conceptually straightforward—at least as long as the outcomes and their probabilities are known to the agent and can be succinctly represented. Agents simply need to choose the course of action that maximizes expected utility. However, things can

| | C | D |
|---|---|---|
| C | −1, −1 | −4, 0 |
| D | 0, −4 | −3, −3 |

**Figure 3.3** The TCP user's (aka the Prisoner's) Dilemma.

get considerably more complicated when the world contains *two or more* utility-maximizing agents whose actions can affect each other's utilities. (To augment our example from Section 3.1.1, what if Bob hates Alice and wants to avoid her too, while Carol is indifferent to seeing Alice and has a crush on Bob? In this case, we might want to revisit our previous assumption that Bob and Carol will act randomly without caring about what the other two agents do.) To study such settings, we turn to game theory.

### 3.2.1 *Example: the TCP user's game*

TCP user's game

Let us begin with a simpler example to provide some intuition about the type of phenomena we would like to study. Imagine that you and another colleague are the only people using the internet. Internet traffic is governed by the TCP protocol. One feature of TCP is the *backoff* mechanism; if the rates at which you and your colleague send information packets into the network causes congestion, you each back off and reduce the rate for a while until the congestion subsides. This is how a correct implementation works. A defective one, however, will not back off when congestion occurs. You have two possible strategies: $C$ (for using a correct implementation) and $D$ (for using a defective one). If both you and your colleague adopt $C$ then your average packet delay is 1 ms. If you both adopt $D$ the delay is 3 ms, because of additional overhead at the network router. Finally, if one of you adopts $D$ and the other adopts $C$ then the $D$ adopter will experience no delay at all, but the $C$ adopter will experience a delay of 4 ms.

These consequences are shown in Figure 3.3. Your options are the two rows, and your colleague's options are the columns. In each cell, the first number represents your payoff (or, the negative of your delay) and the second number represents your colleague's payoff.[1]

Prisoner's Dilemma game

Given these options what should you adopt, $C$ or $D$? Does it depend on what you think your colleague will do? Furthermore, from the perspective of the network operator, what kind of behavior can he expect from the two users? Will any two users behave the same when presented with this scenario? Will the behavior change if the network operator allows the users to communicate with each other before making a decision? Under what changes to the delays would the users' decisions still be the same? How would the users behave if they have

1. A more standard name for this game is the Prisoner's Dilemma; we return to this in Section 3.2.3.

the opportunity to face this same decision with the same counterpart multiple times? Do answers to these questions depend on how rational the agents are and how they view each other's rationality?

Game theory gives answers to many of these questions. It tells us that any rational user, when presented with this scenario once, will adopt $D$—regardless of what the other user does. It tells us that allowing the users to communicate beforehand will not change the outcome. It tells us that for perfectly rational agents, the decision will remain the same even if they play multiple times; however, if the number of times that the agents will play is infinite, or even uncertain, we may see them adopt $C$.

### 3.2.2 *Definition of games in normal form*

The normal form, also known as the strategic form, is the most familiar representation of strategic interactions in game theory. A game written in this way amounts to a representation of every player's utility for every state of the world, in the special case where states of the world depend only on the players' combined actions. Consideration of this special case may seem uninteresting. However, it turns out that settings in which the state of the world also depends on randomness in the environment—called Bayesian games and introduced in Section 6.3—can be reduced to (much larger) normal-form games. Indeed, there also exist normal-form reductions for other game representations, such as games that involve an element of time (extensive-form games, introduced in Chapter 5). Because most other representations of interest can be reduced to it, the normal-form representation is arguably the most fundamental in game theory.

**Definition 3.2.1 (Normal-form game)** *A (finite, n-person)* normal-form game *is a tuple* $(N, A, u)$*, where:*

action
action profile

- *$N$ is a finite set of $n$ players, indexed by $i$;*
- *$A = A_1 \times \cdots \times A_n$, where $A_i$ is a finite set of* actions *available to player $i$. Each vector $a = (a_1, \ldots, a_n) \in A$ is called an* action profile*;*
- *$u = (u_1, \ldots, u_n)$, where $u_i : A \mapsto \mathbb{R}$ is a real-valued* utility *(or* payoff*)* function *for player $i$.*

Note that we previously argued that utility functions should map from the set of *outcomes*, not the set of *actions*. Here we make the implicit assumption that $O = A$.

A natural way to represent games is via an $n$-dimensional matrix. We already saw a two-dimensional example in Figure 3.3. In general, each row corresponds to a possible action for player 1, each column corresponds to a possible action for player 2, and each cell corresponds to one possible outcome. Each player's utility for an outcome is written in the cell corresponding to that outcome, with player 1's utility listed first.

| | $C$ | $D$ |
|---|---|---|
| $C$ | $a, a$ | $b, c$ |
| $D$ | $c, b$ | $d, d$ |

**Figure 3.4** Any $c > a > d > b$ define an instance of Prisoner's Dilemma.

### 3.2.3 *More examples of normal-form games*

#### Prisoner's Dilemma

Previously, we saw an example of a game in normal form, namely, the Prisoner's (or the TCP user's) Dilemma. However, as discussed in Section 3.1.2, the precise payoff numbers play a limited role. The essence of the Prisoner's Dilemma example would not change if the $-4$ was replaced by $-5$, or if 100 was added to each of the numbers. In its most general form, the Prisoner's Dilemma is any normal-form game shown in Figure 3.4, in which $c > a > d > b$.[2]

Incidentally, the name "Prisoner's Dilemma" for this famous game-theoretic situation derives from the original story accompanying the numbers. The players of the game are two prisoners suspected of a crime rather than two network users. The prisoners are taken to separate interrogation rooms, and each can either "confess" to the crime or "deny" it (or, alternatively, "cooperate" or "defect"). If the payoff are all nonpositive, their absolute values can be interpreted as the length of jail term each of prisoner gets in each scenario.

#### Common-payoff games

There are some restricted classes of normal-form games that deserve special mention. The first is the class of *common-payoff games*. These are games in which, for every action profile, all players have the same payoff.

common-payoff game

**Definition 3.2.2 (Common-payoff game)** *A* common-payoff game *is a game in which for all action profiles $a \in A_1 \times \cdots \times A_n$ and any pair of agents $i$, $j$, it is the case that $u_i(a) = u_j(a)$.*

pure coordination game

team game

Common-payoff games are also called *pure coordination games* or *team games*. In such games the agents have no conflicting interests; their sole challenge is to coordinate on an action that is maximally beneficial to all.

As an example, imagine two drivers driving towards each other in a country having no traffic rules, and who must independently decide whether to drive on

2. Under some definitions, there is the further requirement that $a > \frac{b+c}{2}$, which guarantees that the outcome $(C, C)$ maximizes the sum of the agents' utilities.

| | Left | Right |
|---|---|---|
| Left | 1, 1 | 0, 0 |
| Right | 0, 0 | 1, 1 |

**Figure 3.5** Coordination game.

the left or on the right. If the drivers choose the same side (left or right) they have some high utility, and otherwise they have a low utility. The game matrix is shown in Figure 3.5.

### Zero-sum games

zero-sum game

constant-sum game

At the other end of the spectrum from pure coordination games lie *zero-sum games*, which (bearing in mind the comment we made earlier about positive affine transformations) are more properly called *constant-sum games*. Unlike common-payoff games, constant-sum games are meaningful primarily in the context of two-player (though not necessarily two-strategy) games.

**Definition 3.2.3 (Constant-sum game)** *A two-player normal-form game is* constant-sum *if there exists a constant $c$ such that for each strategy profile $a \in A_1 \times A_2$ it is the case that $u_1(a) + u_2(a) = c$.*

For convenience, when we talk of constant-sum games going forward we will always assume that $c = 0$, that is, that we have a zero-sum game. If common-payoff games represent situations of pure coordination, zero-sum games represent situations of pure competition; one player's gain must come at the expense of the other player. This property requires that there be exactly two agents. Indeed, if you allow more agents, any game can be turned into a zero-sum game by adding a dummy player whose actions do not impact the payoffs to the other agents, and whose own payoffs are chosen to make the payoffs in each outcome sum to zero.

Matching Pennies game

A classical example of a zero-sum game is the game of *Matching Pennies*. In this game, each of the two players has a penny and independently chooses to display either heads or tails. The two players then compare their pennies. If they are the same then player 1 pockets both, and otherwise player 2 pockets them. The payoff matrix is shown in Figure 3.6.

The popular children's game of Rock, Paper, Scissors, also known as Rochambeau, provides a three-strategy generalization of the matching-pennies game. The payoff matrix of this zero-sum game is shown in Figure 3.7. In this game, each of the two players can choose either rock, paper, or scissors. If both players choose the same action, there is no winner and the utilities are zero. Otherwise,

| | Heads | Tails |
|---|---|---|
| Heads | $1, -1$ | $-1, 1$ |
| Tails | $-1, 1$ | $1, -1$ |

**Figure 3.6** Matching Pennies game.

| | Rock | Paper | Scissors |
|---|---|---|---|
| Rock | $0, 0$ | $-1, 1$ | $1, -1$ |
| Paper | $1, -1$ | $0, 0$ | $-1, 1$ |
| Scissors | $-1, 1$ | $1, -1$ | $0, 0$ |

**Figure 3.7** Rock, Paper, Scissors game.

each of the actions wins over one of the other actions and loses to the other remaining action.

#### Battle of the Sexes

In general, games can include elements of both coordination and competition. Prisoner's Dilemma does, although in a rather paradoxical way. Here is another well-known game that includes both elements. In this game, called *Battle of the Sexes*, a husband and wife wish to go to the movies, and they can select among two movies: "Lethal Weapon (LW)" and "Wondrous Love (WL)." They much prefer to go together rather than to separate movies, but while the wife (player 1) prefers LW, the husband (player 2) prefers WL. The payoff matrix is shown in Figure 3.8. We will return to this game shortly.

Battle of the Sexes game

### 3.2.4 *Strategies in normal-form games*

We have so far defined the actions available to each player in a game, but not yet his set of *strategies* or his available choices. Certainly one kind of strategy is to select a single action and play it. We call such a strategy a *pure strategy*, and we will use the notation we have already developed for actions to represent it. We call a choice of pure strategy for each agent a *pure-strategy profile*.

pure strategy

pure-strategy profile

| | | Husband: LW | Husband: WL |
|---|---|---|---|
| Wife | LW | 2, 1 | 0, 0 |
| Wife | WL | 0, 0 | 1, 2 |

**Figure 3.8** Battle of the Sexes game.

Players could also follow another, less obvious type of strategy: randomizing over the set of available actions according to some probability distribution. Such a strategy is called a mixed strategy. Although it may not be immediately obvious why a player should introduce randomness into his choice of action, in fact in a multiagent setting the role of mixed strategies is critical. We define a mixed strategy for a normal-form game as follows.

mixed strategy

**Definition 3.2.4 (Mixed strategy)** *Let* $(N, A, u)$ *be a normal-form game, and for any set* $X$ *let* $\Pi(X)$ *be the set of all probability distributions over* $X$*. Then the set of* mixed strategies *for player* $i$ *is* $S_i = \Pi(A_i)$.

mixed-strategy profile

**Definition 3.2.5 (Mixed-strategy profile)** *The set of* mixed-strategy profiles *is simply the Cartesian product of the individual mixed-strategy sets,* $S_1 \times \cdots \times S_n$.

By $s_i(a_i)$ we denote the probability that an action $a_i$ will be played under mixed strategy $s_i$. The subset of actions that are assigned positive probability by the mixed strategy $s_i$ is called the *support* of $s_i$.

support of a mixed strategy

**Definition 3.2.6 (Support)** *The* support *of a mixed strategy* $s_i$ *for a player* $i$ *is the set of pure strategies* $\{a_i | s_i(a_i) > 0\}$.

fully mixed strategy

Note that a pure strategy is a special case of a mixed strategy, in which the support is a single action. At the other end of the spectrum we have *fully mixed strategies*. A strategy is fully mixed if it has full support (i.e., if it assigns every action a nonzero probability).

expected utility

We have not yet defined the payoffs of players given a particular strategy profile, since the payoff matrix defines those directly only for the special case of pure-strategy profiles. But the generalization to mixed strategies is straightforward, and relies on the basic notion of decision theory—*expected utility*. Intuitively, we first calculate the probability of reaching each outcome given the strategy profile, and then we calculate the average of the payoffs of the outcomes, weighted by the probabilities of each outcome. Formally, we define the expected utility as follows (overloading notation, we use $u_i$ for both utility and expected utility).

**Definition 3.2.7 (Expected utility of a mixed strategy)** *Given a normal-form game $(N, A, u)$, the expected utility $u_i$ for player $i$ of the mixed-strategy profile $s = (s_1, \ldots, s_n)$ is defined as*

$$u_i(s) = \sum_{a \in A} u_i(a) \prod_{j=1}^{n} s_j(a_j).$$

## 3.3 Analyzing games: from optimality to equilibrium

Now that we have defined what games in normal form are and what strategies are available to players in them, the question is how to reason about such games. In single-agent decision theory the key notion is that of an *optimal strategy*, that is, a strategy that maximizes the agent's expected payoff for a given environment in which the agent operates. The situation in the single-agent case can be fraught with uncertainty, since the environment might be stochastic, partially observable, and spring all kinds of surprises on the agent. However, the situation is even more complex in a multiagent setting. In this case the environment includes—or, in many cases we discuss, consists entirely of—other agents, all of whom are also hoping to maximize their payoffs. Thus the notion of an optimal strategy for a given agent is not meaningful; the best strategy depends on the choices of others.

optimal strategy

Game theorists deal with this problem by identifying certain subsets of outcomes, called *solution concepts*, that are interesting in one sense or another. In this section we describe two of the most fundamental solution concepts: Pareto optimality and Nash equilibrium.

solution concept

### 3.3.1 *Pareto optimality*

First, let us investigate the extent to which a notion of optimality can be meaningful in games. From the point of view of an outside observer, can some outcomes of a game be said to be better than others?

This question is complicated because we have no way of saying that one agent's interests are more important than another's. For example, it might be tempting to say that we should prefer outcomes in which the sum of agents' utilities is higher. However, recall from Section 3.1.2 that we can apply any positive affine transformation to an agent's utility function and obtain another valid utility function. For example, we could multiply all of player 1's payoffs by 1,000, which could clearly change which outcome maximized the sum of agents' utilities.

Thus, our problem is to find a way of saying that some outcomes are better than others, even when we only know agents' utility functions up to a positive affine transformation. Imagine that each agent's utility is a monetary payment that you will receive, but that each payment comes in a different currency, and you do not know anything about the exchange rates. Which outcomes should you prefer? Observe that, while it is not usually possible to identify the best outcome, there *are* situations in which you can be sure that one outcome is better than another. For example, it is better to get 10 units of currency $A$ and 3 units of currency $B$ than to get 9 units of currency $A$ and 3 units of currency $B$, regardless of the exchange rate. We formalize this intuition in the following definition.

Pareto domination

**Definition 3.3.1 (Pareto domination)** *Strategy profile $s$* Pareto dominates *strategy profile $s'$ if for all $i \in N$, $u_i(s) \geq u_i(s')$, and there exists some $j \in N$ for which $u_j(s) > u_j(s')$.*

In other words, in a Pareto-dominated strategy profile some player can be made better off without making any other player worse off. Observe that we define Pareto domination over strategy profiles, not just action profiles. Thus, here we treat strategy profiles as outcomes, just as we treated lotteries as outcomes in Section 3.1.2.

Pareto domination gives us a partial ordering over strategy profiles. Thus, in answer to our question before, we cannot generally identify a single "best" outcome; instead, we may have a set of noncomparable optima.

Pareto optimality

strict Pareto efficiency

**Definition 3.3.2 (Pareto optimality)** *Strategy profile $s$ is* Pareto optimal, *or* strictly Pareto efficient, *if there does not exist another strategy profile $s' \in S$ that Pareto dominates $s$.*

We can easily draw several conclusions about Pareto optimal strategy profiles. First, every game must have at least one such optimum, and there must always exist at least one such optimum in which all players adopt pure strategies. Second, some games will have multiple optima. For example, in zero-sum games, *all* strategy profiles are strictly Pareto efficient. Finally, in common-payoff games, all Pareto optimal strategy profiles have the same payoffs.

### 3.3.2 *Defining best response and Nash equilibrium*

Now we will look at games from an individual agent's point of view, rather than from the vantage point of an outside observer. This will lead us to the most influential solution concept in game theory, the *Nash equilibrium.*

Our first observation is that if an agent knew how the others were going to play, his strategic problem would become simple. Specifically, he would be left with the single-agent problem of choosing a utility-maximizing action that we discussed in Section 3.1. Formally, define $s_{-i} = (s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n)$, a strategy profile $s$ without agent $i$'s strategy. Thus we can write $s = (s_i, s_{-i})$. If the agents other than $i$ (whom we denote $-i$) were to commit to play $s_{-i}$, a utility-maximizing agent $i$ would face the problem of determining his best response.

best response

**Definition 3.3.3 (Best response)** *Player $i$'s* best response *to the strategy profile $s_{-i}$ is a mixed strategy $s_i^* \in S_i$ such that $u_i(s_i^*, s_{-i}) \geq u_i(s_i, s_{-i})$ for all strategies $s_i \in S_i$.*

The best response is not necessarily unique. Indeed, except in the extreme case in which there is a unique best response that is a pure strategy, the number of best responses is always infinite. When the support of a best response $s^*$ includes two or more actions, the agent must be indifferent among them—otherwise, the agent

| | LW | WL |
|---|---|---|
| LW | 2, 1 | 0, 0 |
| WL | 0, 0 | 1, 2 |

**Figure 3.9** Pure-strategy Nash equilibria in the Battle of the Sexes game.

would prefer to reduce the probability of playing at least one of the actions to zero. But thus *any* mixture of these actions must also be a best response, not only the particular mixture in $s^*$. Similarly, if there are two pure strategies that are individually best responses, any mixture of the two is necessarily also a best response.

Of course, in general an agent will not know what strategies the other players plan to adopt. Thus, the notion of best response is not a solution concept—it does not identify an interesting set of outcomes in this general case. However, we can leverage the idea of best response to define what is arguably the most central notion in noncooperative game theory, the Nash equilibrium.

Nash equilibrium

**Definition 3.3.4 (Nash equilibrium)** *A strategy profile* $s = (s_1, \ldots, s_n)$ *is a* Nash equilibrium *if, for all agents* $i$, $s_i$ *is a best response to* $s_{-i}$.

Intuitively, a Nash equilibrium is a *stable* strategy profile: no agent would want to change his strategy if he knew what strategies the other agents were following.

We can divide Nash equilibria into two categories, strict and weak, depending on whether or not every agent's strategy constitutes a *unique* best response to the other agents' strategies.

strict Nash equilibrium

**Definition 3.3.5 (Strict Nash)** *A strategy profile* $s = (s_1, \ldots, s_n)$ *is a* strict Nash equilibrium *if, for all agents* $i$ *and for all strategies* $s_i' \neq s_i$, $u_i(s_i, s_{-i}) > u_i(s_i', s_{-i})$.

weak Nash equilibrium

**Definition 3.3.6 (Weak Nash)** *A strategy profile* $s = (s_1, \ldots, s_n)$ *is a* weak Nash equilibrium *if, for all agents* $i$ *and for all strategies* $s_i' \neq s_i$, $u_i(s_i, s_{-i}) \geq u_i(s_i', s_{-i})$, *and* $s$ *is not a strict Nash equilibrium.*

Intuitively, weak Nash equilibria are less stable than strict Nash equilibria, because in the former case at least one player has a best response to the other players' strategies that is not his equilibrium strategy. Mixed-strategy Nash equilibria are necessarily weak, while pure-strategy Nash equilibria can be either strict or weak, depending on the game.

### 3.3.3 *Finding Nash equilibria*

Consider again the Battle of the Sexes game. We immediately see that it has two pure-strategy Nash equilibria, depicted in Figure 3.9.

We can check that these are Nash equilibria by confirming that whenever one of the players plays the given (pure) strategy, the other player would only lose by deviating.

| | Heads | Tails |
|---|---|---|
| Heads | $1, -1$ | $-1, 1$ |
| Tails | $-1, 1$ | $1, -1$ |

**Figure 3.10** The Matching Pennies game.

Are these the only Nash equilibria? The answer is no; although they are indeed the only pure-strategy equilibria, there is also another mixed-strategy equilibrium. In general, it is tricky to compute a game's mixed-strategy equilibria; we consider this problem in detail in Chapter 4. However, we will show here that this computational problem is easy when we know (or can guess) the *support* of the equilibrium strategies, particularly so in this small game. Let us now guess that both players randomize, and let us assume that husband's strategy is to play LW with probability $p$ and WL with probability $1 - p$. Then if the wife, the row player, also mixes between her two actions, she must be indifferent between them, given the husband's strategy. (Otherwise, she would be better off switching to a pure strategy according to which she only played the better of her actions.) Then we can write the following equations.

$$U_{\text{wife}}(\text{LW}) = U_{\text{wife}}(\text{WL})$$
$$2 * p + 0 * (1 - p) = 0 * p + 1 * (1 - p)$$
$$p = \frac{1}{3}$$

We get the result that in order to make the wife indifferent between her actions, the husband must choose LW with probability $1/3$ and WL with probability $2/3$. Of course, since the husband plays a mixed strategy he must also be indifferent between his actions. By a similar calculation it can be shown that to make the husband indifferent, the wife must choose LW with probability $2/3$ and WL with probability $1/3$. Now we can confirm that we have indeed found an equilibrium: since both players play in a way that makes the other indifferent, they are both best responding to each other. Like all mixed-strategy equilibria, this is a weak Nash equilibrium. The expected payoff of both agents is $2/3$ in this equilibrium, which means that each of the pure-strategy equilibria Pareto-dominates the mixed-strategy equilibrium.

Earlier, we mentioned briefly that mixed strategies play an important role. The previous example may not make it obvious, but now consider again the Matching Pennies game, reproduced in Figure 3.10. It is not hard to see that no pure strategy could be part of an equilibrium in this game of pure competition. Therefore, likewise there can be no strict Nash equilibrium in this game. But using the aforementioned procedure, the reader can verify that again there exists a mixed-strategy equilibrium; in this case, each player chooses one of the two available actions with probability $1/2$.

What does it mean to say that an agent plays a mixed-strategy Nash equilibrium? Do players really sample probability distributions in their heads? Some people have argued that they really do. One well-known motivating example for mixed strategies involves soccer: specifically, a kicker and a goalie getting ready for a penalty kick. The kicker can kick to the left or the right, and the goalie can jump to the left or the right. The kicker scores if and only if he kicks to one side and the goalie jumps to the other; this is thus best modeled as Matching Pennies. Any pure strategy on the part of either player invites a winning best response on the part of the other player. It is only by kicking or jumping in either direction with equal probability, goes the argument, that the opponent cannot exploit your strategy.

Of course, this argument is not uncontroversial. In particular, it can be argued that the strategies of each player are deterministic, but each player has uncertainty regarding the other player's strategy. This is indeed a second possible interpretation of mixed strategies: the mixed strategy of player $i$ is everyone else's assessment of how likely $i$ is to play each pure strategy. In equilibrium, $i$'s mixed strategy has the further property that every action in its support is a best response to player $i$'s beliefs about the other agents' strategies.

Finally, there are two interpretations that are related to learning in multiagent systems. In one interpretation, the game is actually played many times repeatedly, and the probability of a pure strategy is the fraction of the time it is played in the limit (its so-called *empirical frequency*). In the other interpretation, not only is the game played repeatedly, but each time it involves two different agents selected at random from a large population. In this interpretation, each agent in the population plays a pure strategy, and the probability of a pure strategy represents the fraction of agents playing that strategy. We return to these learning interpretations in Chapter 7.

empirical frequency

### 3.3.4 *Nash's theorem: proving the existence of Nash equilibria*

We have now seen two examples in which we managed to find Nash equilibria (three equilibria for Battle of the Sexes, one equilibrium for Matching Pennies). Did we just luck out? Here there is some good news—it was not just luck. In this section we prove that every game has at least one Nash equilibrium.

First, a disclaimer: this section is more technical than the rest of the chapter. A reader who is prepared to take the existence of Nash equilibria on faith can safely skip to the beginning of Section 3.4 on p. 71. For the bold of heart who remain, we begin with some preliminary definitions.

convexity

**Definition 3.3.7 (Convexity)** *A set $C \subset \mathbb{R}^m$ is* convex *if for every $x, y \in C$ and $\lambda \in [0, 1]$, $\lambda x + (1 - \lambda)y \in C$. For vectors $x^0, \ldots, x^n$ and nonnegative scalars $\lambda_0, \ldots, \lambda_n$ satisfying $\sum_{i=0}^{n} \lambda_i = 1$, the vector $\sum_{i=0}^{n} \lambda_i x^i$ is called a* convex combination *of $x^0, \ldots, x^n$.*

convex combination

For example, a cube is a convex set in $\mathbb{R}^3$; a bowl is not.

affine independence

**Definition 3.3.8 (Affine independence)** *A finite set of vectors* $\{x^0, \ldots, x^n\}$ *in a Euclidean space is* affinely independent *if* $\sum_{i=0}^{n} \lambda_i x^i = 0$ *and* $\sum_{i=0}^{n} \lambda_i = 0$ *imply that* $\lambda_0 = \cdots = \lambda_n = 0$.

An equivalent condition is that $\{x^1 - x^0, x^2 - x^0, \ldots, x^n - x^0\}$ are linearly independent. Intuitively, a set of points is affinely independent if no three points from the set lie on the same line, no four points from the set lie on the same plane, and so on. For example, the set consisting of the origin 0 and the unit vectors $e^1, \ldots, e^n$ is affinely independent.

Next we define a simplex, which is an $n$-dimensional generalization of a triangle.

$n$-simplex

**Definition 3.3.9 ($n$-simplex)** *An* $n$-simplex, *denoted* $x^0 \cdots x^n$, *is the set of all convex combinations of the affinely independent set of vectors* $\{x^0, \ldots, x^n\}$, *that is,*

$$x^0 \cdots x^n = \left\{ \sum_{i=0}^{n} \lambda_i x^i : \forall i \in \{0, \ldots, n\},\ \lambda_i \geq 0;\ and \sum_{i=0}^{n} \lambda_i = 1 \right\}.$$

vertex

$k$-face

Each $x^i$ is called a *vertex* of the simplex $x^0 \cdots x^n$ and each $k$-simplex $x^{i_0} \cdots x^{i_k}$ is called a *k-face* of $x^0 \cdots x^n$, where $i_0, \ldots, i_k \in \{0, \ldots, n\}$. For example, a triangle (i.e., a 2-simplex) has one 2-face (itself), three 1-faces (its sides) and three 0-faces (its vertices).

**Definition 3.3.10 (Standard $n$-simplex)** *The* standard $n$-simplex $\triangle_n$ *is*

$$\left\{ y \in \mathbb{R}^{n+1} : \sum_{i=0}^{n} y_i = 1, \forall i = 0, \ldots, n,\ y_i \geq 0 \right\}.$$

In other words, the standard $n$-simplex is the set of all convex combinations of the $n+1$ unit vectors $e^0, \ldots, e^n$.

simplicial subdivision

**Definition 3.3.11 (Simplicial subdivision)** *A* simplicial subdivision *of an* $n$*-simplex* $T$ *is a finite set of simplexes* $\{T_i\}$ *for which* $\bigcup_{T_i \in T} T_i = T$, *and for any* $T_i, T_j \in T$, $T_i \cap T_j$ *is either empty or equal to a common face.*

Intuitively, this means that a simplex is divided up into a set of smaller simplexes that together occupy exactly the same region of space and that overlap only on their boundaries. Furthermore, when two of them overlap, the intersection must be an entire face of both subsimplexes. Figure 3.11 (left) shows a 2-simplex subdivided into 16 subsimplexes.

Let $y \in x^0 \cdots x^n$ denote an arbitrary point in a simplex. This point can be written as a convex combination of the vertices: $y = \sum_i \lambda_i x^i$. Now define a function that gives the set of vertices "involved" in this point: $\chi(y) = \{i : \lambda_i > 0\}$. We use this function to define a proper labeling.

proper labeling

**Definition 3.3.12 (Proper labeling)** *Let* $T = x^0 \cdots x^n$ *be simplicially subdivided, and let* $V$ *denote the set of all distinct vertices of all the subsimplexes. A function* $\mathcal{L} : V \mapsto \{0, \ldots, n\}$ *is a* proper labeling *of a subdivision if* $\mathcal{L}(v) \in \chi(v)$.

One consequence of this definition is that the vertices of a simplex must all receive different labels. (Do you see why?) As an example, the subdivided simplex in Figure 3.11 (left) is properly labeled.

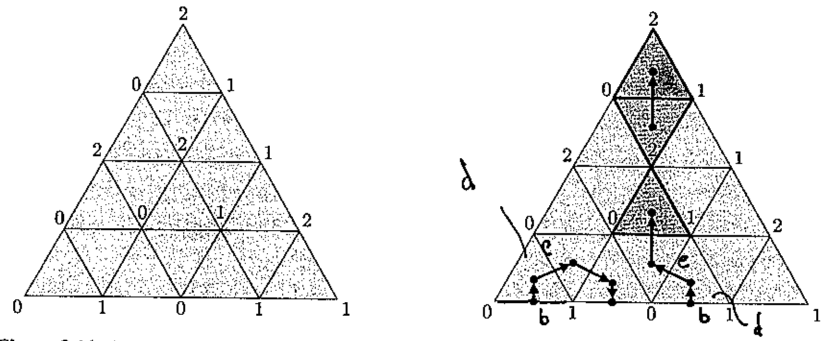

**Figure 3.11** A properly labeled simplex (left), and the same simplex with completely labeled subsimplexes shaded and three walks indicated (right).

completely labeled subsimplex

**Definition 3.3.13 (Complete labeling)** *A subsimplex is* completely labeled *if $\mathcal{L}$ assumes all the values $0, \ldots, n$ on its set of vertices.*

For example in the subdivided triangle in Figure 3.11 (left), the subtriangle at the very top is completely labeled.

Sperner's lemma

**Lemma 3.3.14 (Sperner's lemma)** *Let $T_n = x^0 \cdots x^n$ be simplicially subdivided and let $\mathcal{L}$ be a proper labeling of the subdivision. Then there are an odd number of completely labeled subsimplexes in the subdivision.*

**Proof.** We prove this by induction on $n$. The case $n = 0$ is trivial. The simplex consists of a single point $x^0$. The only possible simplicial subdivision is $\{x^0\}$. There is only one possible labeling function, $\mathcal{L}(x^0) = 0$. Note that this is a proper labeling. So there is one completely labeled subsimplex, $x^0$ itself.

We now assume the statement to be true for $n-1$ and prove it for $n$. The simplicial subdivision of $T_n$ induces a simplicial subdivision on its face $x^0 \cdots x^{n-1}$. This face is an $(n-1)$-simplex; denote it as $T_{n-1}$. The labeling function $\mathcal{L}$ restricted to $T_{n-1}$ is a proper labeling of $T_{n-1}$. Therefore by the induction hypothesis there exist an odd number of $(n-1)$-subsimplexes in $T_{n-1}$ that bear the labels $(0, \ldots, n-1)$. (To provide graphical intuition, we will illustrate the induction argument on a subdivided 2-simplex. In Figure 3.11 (left), observe that the bottom face $x^0x^1$ is a subdivided 1-simplex—a line segment—containing four subsimplexes, three of which are completely labeled.)

We now define rules for "walking" across our subdivided, labeled simplex $T_n$. The walk begins at an $(n-1)$-subsimplex with labels $(0, \ldots, n-1)$ on the face $T_{n-1}$; call this subsimplex $b$. There exists a unique $n$-subsimplex $d$ that has $b$ as a face; $d$'s vertices consist of the vertices of $b$ and another vertex $z$. If $z$ is labeled $n$, then we have a completely labeled subsimplex and the walk ends. Otherwise, $d$ has the labels $(0, \ldots, n-1)$, where one of the labels (say $j$) is repeated, and the label $n$ is missing. In this case there exists exactly one other $(n-1)$-subsimplex that is a face of $d$ and bears the labels $(0, \ldots, n-1)$. This is because each $(n-1)$-face of $d$ is defined by all but one of $d$'s vertices; since only the label $j$ is repeated, an $(n-1)$-face of $d$ has labels $(0, \ldots, n-1)$ if and only if one of the two vertices with label $j$ is left out. We know $b$ is one such face, so there is exactly one other, which we call $e$. (For example, you can confirm in Figure 3.11 (left) that if a subtriangle has an edge with labels (0, 1), then it is either completely labeled, or it has exactly one other edge with labels (0, 1).) We continue the walk from $e$. We make use of the following property: an $(n-1)$-face of an $n$-subsimplex in a simplicially subdivided simplex $T_n$ is either on an $(n-1)$-face of $T_n$, or the intersection of two $n$-subsimplexes. If $e$ is on an $(n-1)$-face of $T_n$ we stop the walk. Otherwise we walk into the unique other $n$-subsimplex having $e$ as a face. This subsimplex is either completely labeled or has one repeated label, and we continue the walk in the same way we did with subsimplex $d$ earlier.

Note that the walk is completely determined by the starting $(n-1)$-subsimplex. The walk ends either at a completely labeled $n$-subsimplex, or at a $(n-1)$-subsimplex with labels $(0, \ldots, n-1)$ on the face $T_{n-1}$. (It cannot end on any other face because $\mathcal{L}$ is a proper labeling.) Note also that every walk can be followed backward: beginning from the end of the walk and following the same rule as earlier, we end up at the starting point. This implies that if a walk starts at $t$ on $T_{n-1}$ and ends at $t'$ on $T_{n-1}$, $t$ and $t'$ must be different, because otherwise we could reverse the walk and get a different path with the same starting point, contradicting the uniqueness of the walk. (Figure 3.11 (right) illustrates one walk of each of the kinds we have discussed so far: one that starts and ends at different subsimplexes on the face $x^0x^1$, and one that starts on the face $x^0x^1$ and ends at a completely labeled subtriangle.) Since by the induction hypothesis there are an odd number of $(n-1)$-subsimplexes with labels $(0, \ldots, n-1)$ at the face $T_{n-1}$, there must be at least one walk that does not end on this face. Since walks that start and end on the face "pair up," there are thus an odd number of walks starting from the face that end at completely labeled subsimplexes. All such walks end at *different* completely labeled subsimplexes, because there is exactly one $(n-1)$-simplex face labeled $(0, \ldots, n-1)$ for a walk to enter from in a completely labeled subsimplex.

Not all completely labeled subsimplexes are led to by such walks. To see why, consider reverse walks starting from completely labeled subsimplexes. Some of these reverse walks end at $(n-1)$-simplexes on $T_{n-1}$, but some end at other completely labeled $n$-subsimplexes. (Figure 3.11 (right) illustrates one walk of this kind.) However, these walks just pair up completely labeled subsimplexes. There are thus an even number of completely labeled subsimplexes that pair up with each other, and an odd number of completely labeled subsimplexes that are led to by walks from the face $T_{n-1}$. Therefore the total number of completely labeled subsimplexes is odd. ■

compactness

**Definition 3.3.15 (Compactness)** *A subset of* $\mathbb{R}^n$ *is* compact *if the set is closed and bounded.*

It is straightforward to verify that $\Delta_m$ is compact. A compact set has the property that every sequence in the set has a convergent subsequence.

centroid

**Definition 3.3.16 (Centroid)** *The* centroid *of a simplex $x^0 \cdots x^m$ is the "average" of its vertices, $\frac{1}{m+1}\sum_{i=0}^{m} x^i$.*

We are now ready to use Sperner's lemma to prove Brouwer's fixed-point theorem.

Brouwer's fixed-point theorem

**Theorem 3.3.17 (Brouwer's fixed-point theorem)** *Let $f : \Delta_m \mapsto \Delta_m$ be continuous. Then $f$ has a fixed point—that is, there exists some $z \in \Delta_m$ such that $f(z) = z$.*

**Proof.** We prove this by first constructing a proper labeling of $\Delta_m$, then showing that as we make finer and finer subdivisions, there exists a subsequence of completely labeled subsimplexes that converges to a fixed point of $f$.

**Part 1: $\mathcal{L}$ is a proper labeling.** Let $\epsilon > 0$. We simplicially subdivide[3] $\Delta_m$ such that the Euclidean distance between any two points in the same $m$-subsimplex is at most $\epsilon$. We define a labeling function $\mathcal{L} : V \mapsto \{0, \ldots, m\}$ as follows. For each $v$ we choose a label satisfying

$$\mathcal{L}(v) \in \chi(v) \cap \{i : f_i(v) \le v_i\}, \tag{3.1}$$

where $v_i$ is the $i^{\text{th}}$ component of $v$ and $f_i(v)$ is the $i^{\text{th}}$ component of $f(v)$. In other words, $\mathcal{L}(v)$ can be any label $i$ such that $v_i > 0$ and $f$ weakly decreases the $i^{\text{th}}$ component of $v$. To ensure that $\mathcal{L}$ is well defined, we must show that the intersection on the right side of Equation (3.1) is always nonempty. (Intuitively, since $v$ and $f(v)$ are both on the standard simplex $\Delta_m$, and on $\Delta_m$ each point's components sum to 1, there must exist a component of $v$ that is weakly decreased by $f$. This intuition holds even though we restrict to the components in $\chi(v)$ because these are exactly all the positive components of $v$.) We now show this formally. For contradiction, assume otherwise. This assumption implies that $f_i(v) > v_i$ for all $i \in \chi(v)$. Recall from the definition of a standard simplex that $\sum_{i=0}^{m} v_i = 1$. Since by the definition of $\chi$, $v_j > 0$ if and only if $j \in \chi(v)$, we have

$$\sum_{j\in\chi(v)} v_j = \sum_{i=0}^{m} v_i = 1. \tag{3.2}$$

Since $f_j(v) > v_j$ for all $j \in \chi(v)$,

$$\sum_{j\in\chi(v)} f_j(v) > \sum_{j\in\chi(v)} v_j = 1. \tag{3.3}$$

But since $f(v)$ is also on the standard simplex $\Delta_m$,

$$\sum_{j\in\chi(v)} f_i(v) \le \sum_{i=0}^{m} f_i(v) = 1. \tag{3.4}$$

Equations (3.3) and (3.4) lead to a contradiction. Therefore, $\mathcal{L}$ is well defined; it is a proper labeling by construction.

3. Here, we implicitly assume that simplices can always be subdivided regardless of dimension. This is true, but surprisingly difficult to show.

**Part 2: As $\epsilon \to 0$, completely labeled subsimplexes converge to fixed points of $f$.** Since $\mathcal{L}$ is a proper labeling, by Sperner's lemma (3.3.14) there is at least one completely labeled subsimplex $p^0 \cdots p^m$ such that $f_i(p^i) \leq p^i_i$ for each $i$. Let $\epsilon \to 0$ and consider the sequence of centroids of completely labeled subsimplexes. Since $\Delta_m$ is compact, there is a convergent subsequence. Let $z$ be its limit; then for all $i = 0, \ldots, m$, $p^i \to z$ as $\epsilon \to 0$. Since $f$ is continuous we must have $f_i(z) \leq z_i$ for all $i$. This implies $f(z) = z$, because otherwise (by an argument similar to the one in Part 1) we would have $1 = \sum_i f_i(z) < \sum_i z_i = 1$, a contradiction. ■

Theorem 3.3.17 cannot be used directly to prove the existence of Nash equilibria. This is because a Nash equilibrium is a point in the set of mixed-strategy profiles $S$. This set is not a simplex but rather a *simplotope*: a Cartesian product of simplexes. (Observe that each individual agent's mixed strategy *can* be understood as a point in a simplex.) However, it turns out that Brouwer's theorem can be extended beyond simplexes to simplotopes.[4] In essence, this is because every simplotope is topologically the same as a simplex (formally, they are *homeomorphic*).

simplotope

**Definition 3.3.18 (Bijective function)** *A function $f$ is* injective *(or one-to-one) if $f(a) = f(b)$ implies $a = b$. A function $f : X \mapsto Y$ is* onto *if for every $y \in Y$ there exists $x \in X$ such that $f(x) = y$. A function is* bijective *if it is both injective and onto.*

bijective

**Definition 3.3.19 (Homeomorphism)** *A set $A$ is* homeomorphic *to a set $B$ if there exists a continuous, bijective function $h : A \mapsto B$ such that $h^{-1}$ is also continuous. Such a function $h$ is called a* homeomorphism.

homeomorphism

**Definition 3.3.20 (Interior)** *A point $x$ is an* interior point *of a set $A \subset \mathbb{R}^m$ if there is an open $m$-dimensional ball $B \subset \mathbb{R}^m$ centered at $x$ such that $B \subset A$. The* interior *of a set $A$ is the set of all its interior points.*

interior

**Corollary 3.3.21 (Brouwer's fixed-point theorem, simplotopes)** *Let $K = \prod_{j=1}^{k} \Delta_{m_j}$ be a simplotope and let $f : K \mapsto K$ be continuous. Then $f$ has a fixed point.*

**Proof.** Let $m = \sum_{j=1}^{k} m_j$. First we show that if $K$ is homeomorphic to $\Delta_m$, then a continuous function $f : K \mapsto K$ has a fixed point. Let $h : \Delta_m \mapsto K$ be a homeomorphism. Then $h^{-1} \circ f \circ h : \Delta_m \mapsto \Delta_m$ is continuous, where $\circ$ denotes function composition. By Theorem 3.3.17 there exists a $z'$ such that $h^{-1} \circ f \circ h(z') = z'$. Let $z = h(z')$, then $h^{-1} \circ f(z) = z' = h^{-1}(z)$. Since $h^{-1}$ is injective, $f(z) = z$.

We must still show that $K = \prod_{j=1}^{k} \Delta_{m_j}$ is homeomorphic to $\Delta_m$. $K$ is convex and compact because each $\Delta_{m_j}$ is convex and compact, and a product of convex and compact sets is also convex and compact. Let the *dimension* of a subset of an Euclidean space be the number of independent parameters

4. An argument similar to our proof below can be used to prove a generalization of Theorem 3.3.17 to arbitrary convex and compact sets.

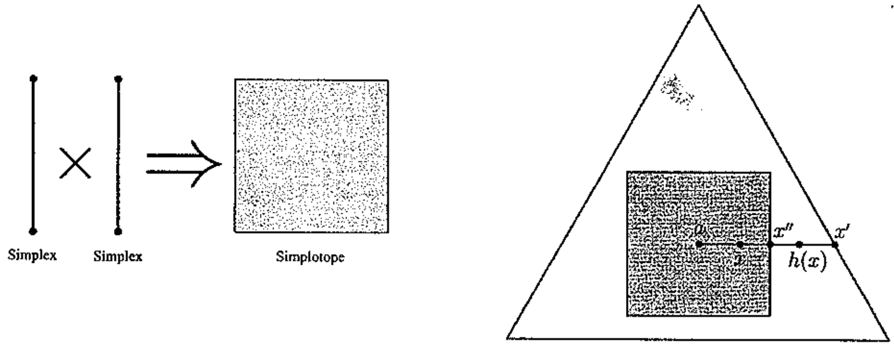


**Figure 3.12** A product of two standard 1-simplexes is a square (a simplotope; left). The square is scaled and put inside a triangle (a 2-simplex), and an example of radial projection $h$ is shown (right).

required to describe each point in the set. For example, an $n$-simplex has dimension $n$. Since each $\Delta_{m_j}$ has dimension $m_j$, $K$ has dimension $m$. Since $K \subset \mathbb{R}^{m+k}$ and $\Delta_m \subset \mathbb{R}^{m+1}$ both have dimension $m$, they can be embedded in $\mathbb{R}^m$ as $K'$ and $\Delta'_m$ respectively. Furthermore, whereas $K \subset \mathbb{R}^{m+k}$ and $\Delta_m \subset \mathbb{R}^{m+1}$ have no interior points, both $K'$ and $\Delta'_m$ have nonempty interior. For example, a standard 2-simplex is defined in $\mathbb{R}^3$, but we can embed the triangle in $\mathbb{R}^2$. As illustrated in Figure 3.12 (left), the product of two standard 1-simplexes is a square, which can also be embedded in $\mathbb{R}^2$. We scale and translate $K'$ into $K''$ such that $K''$ is strictly inside $\Delta'_m$. Since scaling and translation are homeomorphisms, and a chain of homeomorphisms is still a homeomorphism, we just need to find a homeomorphism $h : K'' \mapsto \Delta'_m$. Fix a point $a$ in the interior of $K''$. Define $h$ to be the "radial projection" with respect to $a$, where $h(a) = a$ and for $x \in K'' \setminus \{a\}$,

$$h(x) = a + \frac{||x' - a||}{||x'' - a||}(x - a),$$

where $x'$ is the intersection point of the boundary of $\Delta'_m$ with the ray that starts at $a$ and passes through $x$, and $x''$ is the intersection point of the boundary of $K''$ with the same ray. Because $K''$ and $\Delta'_m$ are convex and compact, $x''$ and $x'$ exist and are unique. Since $a$ is an interior point of $K''$ and $\Delta_m$, $||x' - a||$ and $||x'' - a||$ are both positive. Intuitively, $h$ scales $x$ along the ray by a factor of $\frac{||x'-a||}{||x''-a||}$. Figure 3.12 (right) illustrates an example of this radial projection from a square simplotope to a triangle.

Finally, it remains to show that $h$ is a homeomorphism. It is relatively straightforward to verify that $h$ is continuous. Since we know that $h(x)$ lies on the ray that starts at $a$ and passes through $x$, given $h(x)$ we can reconstruct the same ray by drawing a ray from $a$ that passes through $h(x)$. We can then recover $x'$ and $x''$, and find $x$ by scaling $h(x)$ along the ray by a factor of $\frac{||x''-a||}{||x'-a||}$. Thus $h$ is injective. $h$ is onto because given any point $y \in \Delta'_m$, we can construct the ray and find $x$ such that $h(x) = y$. So, $h^{-1}$ has the same form as $h$ except that the scaling factor is inverted, thus $h^{-1}$ is also continuous. Therefore, $h$ is a homeomorphism. ■

We are now ready to prove the existence of Nash equilibrium. Indeed, now that we have Corollary 3.3.21 and notation for discussing mixed strategies (Section 3.2.4), it is surprisingly easy. The proof proceeds by constructing a continuous $f : S \mapsto S$ such that each fixed point of $f$ is a Nash equilibrium. Then we use Corollary 3.3.21 to argue that $f$ has at least one fixed point, and thus that Nash equilibria always exist.

**Theorem 3.3.22 (Nash, 1951)** *Every game with a finite number of players and action profiles has at least one Nash equilibrium.*

**Proof.** Given a strategy profile $s \in S$, for all $i \in N$ and $a_i \in A_i$ we define

$$\varphi_{i,a_i}(s) = \max\{0, u_i(a_i, s_{-i}) - u_i(s)\}.$$

We then define the function $f : S \mapsto S$ by $f(s) = s'$, where

$$s'_i(a_i) = \frac{s_i(a_i) + \varphi_{i,a_i}(s)}{\sum_{b_i \in A_i} s_i(b_i) + \varphi_{i,b_i}(s)} = \frac{s_i(a_i) + \varphi_{i,a_i}(s)}{1 + \sum_{b_i \in A_i} \varphi_{i,b_i}(s)}. \tag{3.5}$$

Intuitively, this function maps a strategy profile $s$ to a new strategy profile $s'$ in which each agent's actions that are better responses to $s$ receive increased probability mass.

The function $f$ is continuous since each $\varphi_{i,a_i}$ is continuous. Since $S$ is convex and compact and $f : S \mapsto S$, by Corollary 3.3.21 $f$ must have at least one fixed point. We must now show that the fixed points of $f$ are the Nash equilibria.

First, if $s$ is a Nash equilibrium then all $\varphi$'s are 0, making $s$ a fixed point of $f$.

Conversely, consider an arbitrary fixed point of $f$, $s$. By the linearity of expectation there must exist at least one action in the support of $s$, say $a'_i$, for which $u_{i,a'_i}(s) \le u_i(s)$. From the definition of $\varphi$, $\varphi_{i,a'_i}(s) = 0$. Since $s$ is a fixed point of $f$, $s'_i(a'_i) = s_i(a'_i)$. Consider Equation (3.5), the expression defining $s'_i(a'_i)$. The numerator simplifies to $s_i(a'_i)$, and is positive since $a'_i$ is in $i$'s support. Hence the denominator must be 1. Thus for any $i$ and $b_i \in A_i$, $\varphi_{i,b_i}(s)$ must equal 0. From the definition of $\varphi$, this can occur only when no player can improve his expected payoff by moving to a pure strategy. Therefore, $s$ is a Nash equilibrium. ■

## 3.4 Further solution concepts for normal-form games

solution concept

As described earlier at the beginning of Section 3.3, we reason about multiplayer games using *solution concepts*, principles according to which we identify interesting subsets of the outcomes of a game. While the most important solution concept is the Nash equilibrium, there are also a large number of others, only some of which we will discuss here. Some of these concepts are more restrictive than the Nash equilibrium, some less so, and some noncomparable. In Chapters 5 and 6 we will introduce some additional solution concepts that are only applicable to game representations other than the normal form.

### 3.4.1 *Maxmin and minmax strategies*

security level

The *maxmin strategy* of player $i$ in an $n$-player, general-sum game is a (not necessarily unique, and in general mixed) strategy that maximizes $i$'s worst-case payoff, in the situation where all the other players happen to play the strategies which cause the greatest harm to $i$. The *maxmin value* (or *security level*) of the game for player $i$ is that minimum amount of payoff guaranteed by a maxmin strategy.

maxmin strategy
maxmin value

**Definition 3.4.1 (Maxmin)** *The* maxmin strategy *for player $i$ is* $\arg\max_{s_i} \min_{s_{-i}} u_i(s_i, s_{-i})$, *and the* maxmin value *for player $i$ is* $\max_{s_i} \min_{s_{-i}} u_i(s_i, s_{-i})$.

Although the maxmin strategy is a concept that makes sense in simultneous-move games, it can be understood through the following temporal intuition. The maxmin strategy is $i$'s best choice when first $i$ must commit to a (possibly mixed) strategy, and then the remaining agents $-i$ observe this strategy (but not $i$'s action choice) and choose their own strategies to minimize $i$'s expected payoff. In the Battle of the Sexes game (Figure 3.8), the maxmin value for either player is 2/3, and requires the maximizing agent to play a mixed strategy. (Do you see why?)

While it may not seem reasonable to assume that the other agents would be solely interested in minimizing $i$'s utility, it is the case that if $i$ plays a maxmin strategy and the other agents play arbitrarily, $i$ will still receive an expected payoff of at least his maxmin value. This means that the maxmin strategy is a sensible choice for a conservative agent who wants to maximize his expected utility without having to make any assumptions about the other agents, such as that they will act rationally according to their own interests, or that they will draw their action choices from known distributions.

The *minmax strategy* and *minmax value* play a dual role to their maxmin counterparts. In two-player games the minmax strategy for player $i$ against player $-i$ is a strategy that keeps the maximum payoff of $-i$ at a minimum, and the minmax value of player $-i$ is that minimum. This is useful when we want to consider the amount that one player can punish another without regard for his own payoff. Such punishment can arise in repeated games, as we will see in Section 6.1. The formal definitions follow.

minmax strategy
minmax value

**Definition 3.4.2 (Minmax, two-player)** *In a two-player game, the* minmax strategy *for player i against player $-i$ is* $\arg\min_{s_i} \max_{s_{-i}} u_{-i}(s_i, s_{-i})$, *and player $-i$'s* minmax value *is* $\min_{s_i} \max_{s_{-i}} u_{-i}(s_i, s_{-i})$.

In $n$-player games with $n > 2$, defining player $i$'s minmax strategy against player $j$ is a bit more complicated. This is because $i$ will not usually be able to guarantee that $j$ achieves minimal payoff by acting unilaterally. However, if we assume that all the players other than $j$ choose to "gang up" on $j$—and that they are able to coordinate appropriately when there is more than one strategy profile that would yield the same minimal payoff for $j$—then we can define minmax strategies for the $n$-player case.

minmax strategy

**Definition 3.4.3 (Minmax, $n$-player)** *In an $n$-player game, the* minmax strategy *for player $i$ against player $j \neq i$ is $i$'s component of the mixed-strategy profile $s_{-j}$ in the expression* $\arg\min_{s_{-j}} \max_{s_j} u_j(s_j, s_{-j})$, *where $-j$ denotes the set of players other than $j$. As before, the* minmax value *for player $j$ is* $\min_{s_{-j}} \max_{s_j} u_j(s_j, s_{-j})$.

As with the maxmin value, we can give temporal intuition for the minmax value. Imagine that the agents $-i$ must commit to a (possibly mixed) strategy profile, to which $i$ can then play a best response. Player $i$ receives his minmax value if players $-i$ choose their strategies in order to minimize $i$'s expected utility after he plays his best response.

In two-player games, a player's minmax value is always equal to his maxmin value. For games with more than two players a weaker condition holds: a player's maxmin value is always less than or equal to his minmax value. (Can you explain why this is?)

Since neither an agent's maxmin strategy nor his minmax strategy depend on the strategies that the other agents actually choose, the maxmin and minmax strategies give rise to solution concepts in a straightforward way. We will call a mixed-strategy profile $s = (s_1, s_2, \ldots)$ a *maxmin strategy profile* of a given game if $s_1$ is a maxmin strategy for player 1, $s_2$ is a maxmin strategy for player 2 and so on. In two-player games, we can also define *minmax strategy profiles* analogously. In two-player, zero-sum games, there is a very tight connection between minmax and maxmin strategy profiles. Furthermore, these solution concepts are also linked to the Nash equilibrium.

**Theorem 3.4.4 (Minimax theorem (von Neumann, 1928))** *In any finite, two-player, zero-sum game, in any Nash equilibrium[5] each player receives a payoff that is equal to both his maxmin value and his minmax value.*

**Proof.** At least one Nash equilibrium must exist by Theorem 3.3.22. Let $(s'_i, s'_{-i})$ be an arbitrary Nash equilibrium, and denote $i$'s equilibrium payoff as $v_i$. Denote $i$'s maxmin value as $\bar{v}_i$ and $i$'s minmax value as $\underline{v}_i$.

First, show that $\bar{v}_i = v_i$. Clearly we cannot have $\bar{v}_i > v_i$, as if this were true then $i$ would profit by deviating from $s'_i$ to his maxmin strategy, and hence $(s'_i, s'_{-i})$ would not be a Nash equilibrium. Thus it remains to show that $\bar{v}_i$ cannot be less than $v_i$.

Assume that $\bar{v}_i < v_i$. By definition, in equilibrium each player plays a best response to the other. Thus

$$v_{-i} = \max_{s_{-i}} u_{-i}(s'_i, s_{-i}).$$

5. The attentive reader might wonder how a theorem from 1928 can use the term "Nash equilibrium," when Nash's work was published in 1950. Von Neumann used different terminology and proved the theorem in a different way; however, the given presentation is probably clearer in the context of modern game theory.

Equivalently, we can write that $-i$ minimizes the negative of his payoff, given $i$'s strategy,

$$-v_{-i} = \min_{s_{-i}} -u_{-i}(s_i', s_{-i}).$$

Since the game is zero sum, $v_i = -v_{-i}$ and $u_i = -u_{-i}$. Thus,

$$v_i = \min_{s_{-i}} u_i(s_i', s_{-i}).$$

We defined $\bar{v}_i$ as $\max_{s_i} \min_{s_{-i}} u_i(s_i, s_{-i})$. By the definition of max, we must have

$$\max_{s_i} \min_{s_{-i}} u_i(s_i, s_{-i}) \geq \min_{s_{-i}} u_i(s_i', s_{-i}).$$

Thus $\bar{v}_i \geq v_i$, contradicting our assumption.

We have shown that $\bar{v}_i = v_i$. The proof that $\underline{v}_i = v_i$ is similar, and is left as an exercise. ■

Why is the minmax theorem important? It demonstrates that maxmin strategies, minmax strategies and Nash equilibria coincide in two-player, zero-sum games. In particular, Theorem 3.4.4 allows us to conclude that in two-player, zero-sum games:

value of a zero-sum game

1. Each player's maxmin value is equal to his minmax value. By convention, the maxmin value for player 1 is called the *value of the game*;
2. For both players, the set of maxmin strategies coincides with the set of minmax strategies; and
3. Any maxmin strategy profile (or, equivalently, minmax strategy profile) is a Nash equilibrium. Furthermore, these are all the Nash equilibria. Consequently, all Nash equilibria have the same payoff vector (namely, those in which player 1 gets the value of the game).

For example, in the Matching Pennies game in Figure 3.6, the value of the game is 0. The unique Nash equilibrium consists of both players randomizing between heads and tails with equal probability, which is both the maxmin strategy and the minmax strategy for each player.

Nash equilibria in zero-sum games can be viewed graphically as a "saddle" in a high-dimensional space. At a saddle point, any deviation of the agent lowers his utility and increases the utility of the other agent. It is easy to visualize in the simple case in which each agent has two pure strategies. In this case the space of mixed strategy profiles can be viewed as the points on the square between (0,0) and (1,1). Adding a third dimension representing player 1's expected utility, the payoff to player 1 under these mixed strategy profiles (and thus the negative of the payoff to player 2) is a saddle-shaped surface. Figure 3.13 (left) gives a pictorial example, illustrating player 1's expected utility in Matching Pennies as a function of both players' probabilities of playing heads. Figure 3.13 (right) adds a plane at $z = 0$ to make it easier to see that it is an equilibrium for both players to play heads 50% of the time and that zero is both the maxmin value and the minmax value for both players.

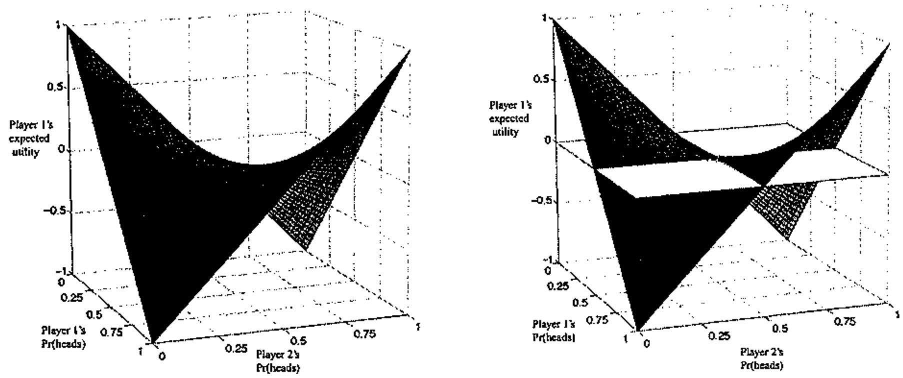


**Figure 3.13** The saddle point in Matching Pennies, with and without a plane at $z = 0$.

| | $L$ | $R$ |
|---|---|---|
| $T$ | $100, a$ | $1 - \epsilon, b$ |
| $B$ | $2, c$ | $1, d$ |

**Figure 3.14** A game for contrasting maxmin with minimax regret. The numbers refer only to player 1's payoffs; $\epsilon$ is an arbitrarily small positive constant. Player 2's payoffs are the arbitrary (and possibly unknown) constants $a, b, c,$ and $d$.

## 3.4.2 *Minimax regret*

We argued earlier that agents might play maxmin strategies in order to achieve good payoffs in the worst case, even in a game that is not zero sum. However, consider a setting in which the other agent is not believed to be malicious, but is instead entirely unpredictable. (Crucially, in this section we do not approach the problem as Bayesians, saying that agent $i$'s beliefs can be described by a probability distribution; instead, we use a "pre-Bayesian" model in which $i$ does not know such a distribution and indeed has no beliefs about it.) In such a setting, it can make sense for agents to care about minimizing their worst-case *losses*, rather than maximizing their worst-case payoffs.

Consider the game in Figure 3.14. Let $\epsilon$ be an arbitrarily small positive constant. For this example it does not matter what agent 2's payoffs $a, b, c,$ and $d$ are, and we can even imagine that agent 1 does not know these values. Indeed, this could be one reason why player 1 would be unable to form beliefs about how player 2 would play, even if he were to believe that player 2 was rational. Let us imagine that agent 1 wants to determine a strategy to follow that makes sense despite his uncertainty about player 2. First, agent 1 might play his maxmin, or "safety level" strategy. In this game it is easy to see that player 1's maxmin strategy is to play $B$; this is because player 2's minmax strategy is to play $R$, and $B$ is a best response to $R$.

If player 1 does not believe that player 2 is malicious, however, he might instead reason as follows. If player 2 were to play $R$ then it would not matter very much how player 1 plays: the most he could lose by playing the wrong way would be $\epsilon$. On the other hand, if player 2 were to play $L$ then player 1's action would be very significant: if player 1 were to make the wrong choice here then his utility would be decreased by 98. Thus player 1 might choose to play $T$ in order to minimize his worst-case loss. Observe that this is the opposite of what he would choose if he followed his maxmin strategy.

Let us now formalize this idea. We begin with the notion of regret.

regret

**Definition 3.4.5 (Regret)** *An agent $i$'s* regret *for playing an action $a_i$ if the other agents adopt action profile $a_{-i}$ is defined as*

$$\left[\max_{a_i' \in A_i} u_i(a_i', a_{-i})\right] - u_i(a_i, a_{-i}).$$

In words, this is the amount that $i$ loses by playing $a_i$, rather than playing his best response to $a_{-i}$. Of course, $i$ does not know what actions the other players will take; however, he can consider those actions that would give him the highest regret for playing $a_i$.

maximum regret

**Definition 3.4.6 (Max regret)** *An agent $i$'s* maximum regret *for playing an action $a_i$ is defined as*

$$\max_{a_{-i} \in A_{-i}} \left(\left[\max_{a_i' \in A_i} u_i(a_i', a_{-i})\right] - u_i(a_i, a_{-i})\right).$$

This is the amount that $i$ loses by playing $a_i$ rather than playing his best response to $a_{-i}$, if the other agents chose the $a_{-i}$ that makes this loss as large as possible. Finally, $i$ can choose his action in order to minimize this worst-case regret.

**Definition 3.4.7 (Minimax regret)** *Minimax regret actions for agent $i$ are defined as*

$$\arg\min_{a_i \in A_i} \left[\max_{a_{-i} \in A_{-i}} \left(\left[\max_{a_i' \in A_i} u_i(a_i', a_{-i})\right] - u_i(a_i, a_{-i})\right)\right].$$

Thus, an agent's minimax regret action is an action that yields the smallest maximum regret. Minimax regret can be extended to a solution concept in the natural way, by identifying action profiles that consist of minimax regret actions for each player. Note that we can safely restrict ourselves to actions rather than mixed strategies in the definitions above (i.e., maximizing over the sets $A_i$ and $A_{-i}$ instead of $S_i$ and $S_{-i}$), because of the linearity of expectation. We leave the proof of this fact as an exercise.

### 3.4.3 *Removal of dominated strategies*

We first define what it means for one strategy to dominate another. Intuitively, one strategy dominates another for a player $i$ if the first strategy yields $i$ a greater payoff than the second strategy, for *any* strategy profile of the remaining players.[6] There are, however, three gradations of dominance, which are captured in the following definition.

**Definition 3.4.8 (Domination)** *Let $s_i$ and $s_i'$ be two strategies of player $i$, and $S_{-i}$ the set of all strategy profiles of the remaining players. Then:*

strict domination

1. $s_i$ strictly dominates $s_i'$ *if for all $s_{-i} \in S_{-i}$, it is the case that $u_i(s_i, s_{-i}) > u_i(s_i', s_{-i})$.*

weak domination

2. $s_i$ weakly dominates $s_i'$ *if for all $s_{-i} \in S_{-i}$, it is the case that $u_i(s_i, s_{-i}) \geq u_i(s_i', s_{-i})$, and for at least one $s_{-i} \in S_{-i}$, it is the case that $u_i(s_i, s_{-i}) > u_i(s_i', s_{-i})$.*

very weak domination

3. $s_i$ very weakly dominates $s_i'$ *if for all $s_{-i} \in S_{-i}$, it is the case that $u_i(s_i, s_{-i}) \geq u_i(s_i', s_{-i})$.*

If one strategy dominates all others, we say that it is (strongly, weakly or very weakly) *dominant.*

**Definition 3.4.9 (Dominant strategy)** *A strategy is* strictly (resp., weakly; very weakly) dominant *for an agent if it strictly (weakly; very weakly) dominates any other strategy for that agent.*

equilibrium in dominant strategies

It is obvious that a strategy profile $(s_1, \ldots, s_n)$ in which every $s_i$ is dominant for player $i$ (whether strictly, weakly, or very weakly) is a Nash equilibrium. Such a strategy profile forms what is called an *equilibrium in dominant strategies* with the appropriate modifier (*strictly*, etc). An equilibrium in strictly dominant strategies is necessarily the unique Nash equilibrium. For example, consider again the Prisoner's Dilemma game. For each player, the strategy $D$ is strictly dominant, and indeed $(D, D)$ is the unique Nash equilibrium. Indeed, we can now explain the "dilemma" which is particularly troubling about the Prisoner's Dilemma game: the outcome reached in the unique equilibrium, which is an equilibrium in strictly dominant strategies, is also the only outcome that is *not* Pareto optimal.

mechanism design

Games with dominant strategies play an important role in game theory, especially in games handcrafted by experts. This is true in particular in *mechanism design*, as we will see in Chapter 10. However, dominant strategies are rare in naturally-occurring games. More common are dominated strategies.

dominated strategy

**Definition 3.4.10 (Dominated strategy)** *A strategy $s_i$ is* strictly (weakly; very weakly) dominated *for an agent $i$ if some other strategy $s_i'$ strictly (weakly; very weakly) dominates $s_i$.*

6. Note that here we consider strategy domination from one individual player's point of view; thus, this notion is unrelated to the concept of Pareto domination discussed earlier.

Let us focus for the moment on strictly dominated strategies. Intuitively, all strictly dominated pure strategies can be ignored, since they can never be best responses to any moves by the other players. There are several subtleties, however. First, once a pure strategy is eliminated, another strategy that was not dominated can become dominated. And so this process of elimination can be continued. Second, a pure strategy may be dominated by a mixture of other pure strategies without being dominated by any of them independently. To see this, consider the game in Figure 3.15.

| | $L$ | $C$ | $R$ |
|---|---|---|---|
| $U$ | 3, 1 | 0, 1 | 0, 0 |
| $M$ | 1, 1 | 1, 1 | 5, 0 |
| $D$ | 0, 1 | 4, 1 | 0, 0 |

**Figure 3.15** A game with dominated strategies.

Column $R$ can be eliminated, since it is dominated by, for example, column $L$. We are left with the reduced game in Figure 3.16.

| | $L$ | $C$ |
|---|---|---|
| $U$ | 3, 1 | 0, 1 |
| $M$ | 1, 1 | 1, 1 |
| $D$ | 0, 1 | 4, 1 |

**Figure 3.16** The game from Figure 3.15 after removing the dominated strategy $R$.

In this game $M$ is dominated by neither $U$ nor $D$, but it is dominated by the mixed strategy that selects either $U$ or $D$ with equal probability. (Note, however, that it was not dominated before the elimination of the $R$ column.) And so we are left with the maximally reduced game in Figure 3.17.

This yields us a solution concept: the set of all strategy profiles that assign zero probability to playing any action that would be removed through iterated removal of strictly dominated strategies. Note that this is a much weaker solution concept than Nash equilibrium—the set of strategy profiles will include all the Nash equilibria, but it will include many other mixed strategies as well. In some games, it will be equal to $S$, the set of all possible mixed strategies.

| | $L$ | $C$ |
|---|---|---|
| $U$ | 3, 1 | 0, 1 |
| $D$ | 0, 1 | 4, 1 |

**Figure 3.17** The game from Figure 3.16 after removing the dominated strategy $M$.

Since iterated removal of strictly dominated strategies preserves Nash equilibria, we can use this technique to computational advantage. In the previous example, rather than computing the Nash equilibria of the original $3 \times 3$ game, we can now compute them for this $2 \times 2$ game, applying the technique described earlier. In some cases, the procedure ends with a single cell; this is the case, for example, with the Prisoner's Dilemma game. In this case we say that the game is *solvable* by iterated elimination.

Clearly, in any finite game, iterated elimination ends after a finite number of iterations. One might worry that, in general, the order of elimination might affect the final outcome. It turns out that this elimination order does not matter when we remove *strictly* dominated strategies. (This is called a *Church–Rosser* property.) However, the elimination order can make a difference to the final reduced game if we remove weakly or very weakly dominated strategies.

Church–Rosser property

Which flavor of domination should we concern ourselves with? In fact, each flavor has advantages and disadvantages, which is why we present all of them here. Strict domination leads to better-behaved iterated elimination: it yields a reduced game that is independent of the elimination order, and iterated elimination is more computationally manageable. (This and other computational issues regarding domination are discussed in Section 4.5.3.) There is also a further related advantage that we will defer to Section 3.4.4. Weak domination can yield smaller reduced games, but under iterated elimination the reduced game can depend on the elimination order. Very weak domination can yield even smaller reduced games, but again these reduced games depend on elimination order. Furthermore, very weak domination does not impose a strict order on strategies: when two strategies are equivalent, each very weakly dominates the other. For this reason, this last form of domination is generally considered the least important.

### 3.4.4 *Rationalizability*

rationalizable strategy

A strategy is *rationalizable* if a perfectly rational player could justifiably play it against one or more perfectly rational opponents. Informally, a strategy profile for player $i$ is rationalizable if it is a best response to some beliefs that $i$ could have about the strategies that the other players will take. The wrinkle, however, is that $i$ cannot have arbitrary beliefs about the other players' actions—his beliefs must take into account his knowledge of *their* rationality, which incorporates their knowledge of *his* rationality, their knowledge of his knowledge of their rationality, and so on in an infinite regress. A rationalizable strategy profile is a strategy profile that consists only of rationalizable strategies.

For example, in the Matching Pennies game given in Figure 3.6, the pure strategy *heads* is rationalizable for the row player. First, the strategy *heads* is a best response to the pure strategy *heads* by the column player. Second, believing that the column player would also play *heads* is consistent with the column player's rationality: the column player could believe that the row player would play *tails*, to which the column player's best response is *heads*. It would be rational for the column player to believe that the row player would play *tails* because the column player could believe that the row player believed that the column player would play *tails*, to which *tails* is a best response. Arguing in the same way, we can make our way up the chain of beliefs.

However, not every strategy can be justified in this way. For example, considering the Prisoner's Dilemma game given in Figure 3.3, the strategy $C$ is not rationalizable for the row player, because $C$ is not a best response to any strategy that the column player could play. Similarly, consider the game from Figure 3.15. $M$ is not a rationalizable strategy for the row player: although it *is* a best response to a strategy of the column player's ($R$), there do not exist any beliefs that the column player could hold about the row player's strategy to which $R$ would be a best response.

Because of the infinite regress, the formal definition of rationalizability is somewhat involved; however, it turns out that there are some intuitive things that we can say about rationalizable strategies. First, Nash equilibrium strategies are always rationalizable: thus, the set of rationalizable strategies (and strategy profiles) is always nonempty. Second, in two-player games rationalizable strategies have a simple characterization: they are those strategies that survive the iterated elimination of strictly dominated strategies. In $n$-player games there exist strategies that survive iterated removal of dominated strategies but are not rationalizable. In this more general case, rationalizable strategies are those strategies that survive iterative removal of strategies that are never a best response to any strategy profile by the other players.

We now define rationalizability more formally. First we will define an infinite sequence of (possibly mixed) strategies $S_i^0, S_i^1, S_i^2, \ldots$ for each player $i$. Let $S_i^0 = S_i$; thus, for each agent $i$, the first element in the sequence is the set of all $i$'s mixed strategies. Let $CH(S)$ denote the convex hull of a set $S$: the smallest convex set containing all the elements of $S$. Now we define $S_i^k$ as the set of all strategies $s_i \in S_i^{k-1}$ for which there exists some $s_{-i} \in \prod_{j \neq i} CH(S_j^{k-1})$ such that for all $s_i' \in S_i^{k-1}$, $u_i(s_i, s_{-i}) \geq u_i(s_i', s_{-i})$. That is, a strategy belongs to $S_i^k$ if there is some strategy $s_{-i}$ for the other players in response to which $s_i$ is at least as good as any other strategy from $S_i^{k-1}$. The convex hull operation allows $i$ to best respond to uncertain beliefs about which strategies from $S_j^{k-1}$ player $j$ will adopt. $CH(S_j^{k-1})$ is used instead of $\Pi(S_j^{k-1})$, the set of all probability distributions over $S_j^{k-1}$, because the latter would allow consideration of mixed strategies that are dominated by some pure strategies for $j$. Player $i$

| | LW | WL |
|---|---|---|
| LW | 2, 1 | 0, 0 |
| WL | 0, 0 | 1, 2 |

**Figure 3.18** Battle of the Sexes game.

could not believe that $j$ would play such a strategy because such a belief would be inconsistent with $i$'s knowledge of $j$'s rationality.

Now we define the set of rationalizable strategies for player $i$ as the intersection of the sets $S_i^0, S_i^1, S_i^2, \ldots$.

rationalizable strategy

**Definition 3.4.11 (Rationalizable strategies)** *The* rationalizable strategies *for player i are* $\bigcap_{k=0}^{\infty} S_i^k$.

### 3.4.5 *Correlated equilibrium*

The correlated equilibrium is a solution concept that generalizes the Nash equilibrium. Some people feel that this is the most fundamental solution concept of all.[7]

In a standard game, each player mixes his pure strategies independently. For example, consider again the Battle of the Sexes game (reproduced here as Figure 3.18) and its mixed-strategy equilibrium.

As we saw in Section 3.3.3, this game's unique mixed-strategy equilibrium yields each player an expected payoff of 2/3. But now imagine that the two players can observe the result of a fair coin flip and can condition their strategies based on that outcome. They can now adopt strategies from a richer set; for example, they could choose "WL if heads, LW if tails." Indeed, this pair forms an equilibrium in this richer strategy space; given that one player plays the strategy, the other player only loses by adopting another. Furthermore, the expected payoff to each player in this so-called correlated equilibrium is $.5 * 2 + .5 * 1 = 1.5$. Thus both agents receive higher utility than they do under the mixed-strategy equilibrium in the uncorrelated case (which had expected payoff of 2/3 for both agents), and the outcome is fairer than either of the pure-strategy equilibria in the sense that the worst-off player achieves higher expected utility. Correlating devices can thus be quite useful.

The aforementioned example had both players observe the exact outcome of the coin flip, but the general setting does not require this. Generally, the setting includes some random variable (the "external event") with a commonly-known probability distribution, and a private signal to each player about the instantiation

7. A Nobel-prize-winning game theorist, R. Myerson, has gone so far as to say that "if there is intelligent life on other planets, in a majority of them, they would have discovered correlated equilibrium before Nash equilibrium."

of the random variable. A player's signal can be correlated with the random variable's value and with the signals received by other players, without uniquely identifying any of them. Standard games can be viewed as the degenerate case in which the signals of the different agents are probabilistically independent.

To model this formally, consider $n$ random variables, with a joint distribution over these variables. Imagine that nature chooses according to this distribution, but reveals to each agent only the realized value of his variable, and that the agent can condition his action on this value.[8]

correlated equilibrium

**Definition 3.4.12 (Correlated equilibrium)** *Given an $n$-agent game $G = (N, A, u)$, a* correlated equilibrium *is a tuple $(\mathbf{D}, \pi, \sigma)$, where $\mathbf{d}$ is a tuple of random variables $\mathbf{d} = (\mathbf{d}_1, \ldots, \mathbf{d}_n)$ with respective domains $D = (D_1, \ldots, D_n)$, $\pi$ is a joint distribution over $\mathbf{d}$, $\sigma = (\sigma_1, \ldots, \sigma_n)$ is a vector of mappings $\sigma_i : D_i \mapsto A_i$, and for each agent $i$ and every mapping $\sigma'_i : D_i \mapsto A_i$ it is the case that*

$$\sum_{d \in D} \pi(d) u_i\left(\sigma_1(d_1), \ldots, \sigma_i(d_i), \ldots, \sigma_n(d_n)\right)$$
$$\geq \sum_{d \in D} \pi(d) u_i\left(\sigma_1(d_1), \ldots, \sigma'_i(d_i), \ldots, \sigma_n(d_n)\right).$$

Note that the mapping is to an action—that is, to a pure strategy rather than a mixed one. One could allow a mapping to mixed strategies, but that would add no greater generality. (Do you see why?)

For every Nash equilibrium, we can construct an equivalent correlated equilibrium, in the sense that they induce the same distribution on outcomes.

**Theorem 3.4.13** *For every Nash equilibrium $\sigma^*$ there exists a corresponding correlated equilibrium $\sigma$.*

The proof is straightforward. Roughly, we can construct a correlated equilibrium from a given Nash equilibrium by letting each $D_i = A_i$ and letting the joint probability distribution be $\pi(d) = \prod_{i \in N} \sigma_i^*(d_i)$. Then we choose $\sigma_i$ as the mapping from each $d_i$ to the corresponding $a_i$. When the agents play the strategy profile $\sigma$, the distribution over outcomes is identical to that under $\sigma^*$. Because the $v_i$'s are uncorrelated and no agent can benefit by deviating from $\sigma^*$, $\sigma$ is a correlated equilibrium.

On the other hand, not every correlated equilibrium is equivalent to a Nash equilibrium; the Battle-of-the-Sexes example given earlier provides a counter-example. Thus, correlated equilibrium is a strictly weaker notion than Nash equilibrium.

Finally, we note that correlated equilibria can be combined together to form new correlated equilibria. Thus, if the set of correlated equilibria of a game $G$ does not contain a single element, it is infinite. Indeed, any convex combination of correlated equilibrium payoffs can itself be realized as the payoff profile of some correlated equilibrium. The easiest way to understand this claim is to imagine

8. This construction is closely related to two other constructions later in the book, one in connection with Bayesian Games in Chapter 6, and one in connection with knowledge and probability (KP) structures in Chapter 13.

a public random device that selects which of the correlated equilibria will be played; next, another random number is chosen in order to allow the chosen equilibrium to be played. Overall, each agent's expected payoff is the weighted sum of the payoffs from the correlated equilibria that were combined. Since no agent has an incentive to deviate regardless of the probabilities governing the first random device, we can achieve any convex combination of correlated equilibrium payoffs. Finally, observe that having two stages of random number generation is not necessary: we can simply derive new domains $D$ and a new joint probability distribution $\pi$ from the $D$'s and $\pi$'s of the original correlated equilibria, and so perform the random number generation in one step.

### 3.4.6 *Trembling-hand perfect equilibrium*

Another important solution concept is the *trembling-hand perfect equilibrium*, or simply *perfect equilibrium*. While rationalizability is a weaker notion than that of a Nash equilibrium, perfection is a stronger one. Several equivalent definitions of the concept exist. In the following definition, recall that a fully mixed strategy is one that assigns every action a strictly positive probability.

trembling-hand perfect equilibrium

**Definition 3.4.14 (Trembling-hand perfect equilibrium)** *A mixed strategy $S$ is a* (trembling-hand) perfect equilibrium *of a normal-form game $G$ if there exists a sequence $S^0, S^1, \ldots$ of fully mixed-strategy profiles such that* $\lim_{n\to\infty} S^n = S$, *and such that for each $S^k$ in the sequence and each player $i$, the strategy $s_i$ is a best response to the strategies $s^k_{-i}$.*

Perfect equilibria are relevant to one aspect of multiagent learning (see Chapter 7), which is why we mention them here. However, we do not discuss them in any detail; they are an involved topic, and relate to other subtle refinements of the Nash equilibrium such as the *proper equilibrium*. The notes at the end of the chapter point the reader to further readings on this topic. We should, however, at least explain the term "trembling hand." One way to think about the concept is as requiring that the equilibrium be robust against slight errors—"trembles"—on the part of players. In other words, one's action ought to be the best response not only against the opponents' equilibrium strategies, but also against small perturbation of those. However, since the mathematical definition speaks about arbitrarily small perturbations, whether these trembles in fact model player fallibility or are merely a mathematical device is open to debate.

proper equilibrium

### 3.4.7 *$\epsilon$-Nash equilibrium*

Our final solution concept reflects the idea that players might not care about changing their strategies to a best response when the amount of utility that they could gain by doing so is very small. This leads us to the idea of an $\epsilon$-Nash equilibrium.

**Definition 3.4.15 ($\epsilon$-Nash)** *Fix $\epsilon > 0$. A strategy profile $s = (s_1, \ldots, s_n)$ is an $\epsilon$-Nash equilibrium if, for all agents $i$ and for all strategies $s'_i \neq s_i$, $u_i(s_i, s_{-i}) \geq u_i(s'_i, s_{-i}) - \epsilon$.*

This concept has various attractive properties. $\epsilon$-Nash equilibria always exist; indeed, every Nash equilibrium is surrounded by a region of $\epsilon$-Nash equilibria for any $\epsilon > 0$. The argument that agents are indifferent to sufficiently small gains is convincing to many. Further, the concept can be computationally useful: algorithms that aim to identify $\epsilon$-Nash equilibria need to consider only a finite set of mixed-strategy profiles rather than the whole continuous space. (Of course, the size of this finite set depends on both $\epsilon$ and on the game's payoffs.) Since computers generally represent real numbers using a floating-point approximation, it is usually the case that even methods for the "exact" computation of Nash equilibria (see e.g., Section 4.2) actually find only $\epsilon$-equilibria where $\epsilon$ is roughly the "machine precision" (on the order of $10^{-16}$ or less for most modern computers). $\epsilon$-Nash equilibria are also important to multiagent learning algorithms; we discuss them in that context in Section 7.3.

However, $\epsilon$-Nash equilibria also have several drawbacks. First, although Nash equilibria are always surrounded by $\epsilon$-Nash equilibria, the reverse is not true. Thus, a given $\epsilon$-Nash equilibrium is not necessarily close to any Nash equilibrium. This undermines the sense in which $\epsilon$-Nash equilibria can be understood as approximations of Nash equilibria. Consider the game in Figure 3.19.

| | $L$ | $R$ |
|---|---|---|
| $U$ | $1, 1$ | $0, 0$ |
| $D$ | $1 + \frac{\epsilon}{2}, 1$ | $500, 500$ |

**Figure 3.19** A game with an interesting $\epsilon$-Nash equilibrium.

This game has a unique Nash equilibrium of $(D, R)$, which can be identified through the iterated removal of dominated strategies. ($D$ dominates $U$ for player 1; on the removal of $U$, $R$ dominates $L$ for player 2.) $(D, R)$ is also an $\epsilon$-Nash equilibrium, of course. However, there is also another $\epsilon$-Nash equilibrium: $(U, L)$. This game illustrates two things.

First, neither player's payoff under the $\epsilon$-Nash equilibrium is within $\epsilon$ of his payoff in a Nash equilibrium; indeed, in general both players' payoffs under an $\epsilon$-Nash equilibrium can be arbitrarily less than in any Nash equilibrium. The problem is that the requirement that player 1 cannot gain more than $\epsilon$ by deviating from the $\epsilon$-Nash equilibrium strategy profile of $(U, L)$ does not imply that *player 2* would not be able to gain more than $\epsilon$ by best responding to player 1's deviation.

Second, some $\epsilon$-Nash equilibria might be very unlikely to arise in play. Although player 1 might not care about a gain of $\frac{\epsilon}{2}$, he might reason that the fact that $D$ dominates $U$ would lead player 2 to expect him to play $D$, and that player 2 would thus play $R$ in response. Player 1 might thus play $D$ because it is his best response to $R$. Overall, the idea of $\epsilon$-approximation is much messier when applied to the identification of a fixed point than when it is applied to a (single-objective) optimization problem.

## 3.5 History and references

There exist several excellent technical introductory textbooks for game theory, including Osborne and Rubinstein [1994], Fudenberg and Tirole [1991], and Myerson [1991]. The reader interested in gaining deeper insight into game theory should consult not only these, but also the most relevant strands of the the vast literature on game theory which has evolved over the years.

The origins of the material covered in the chapter are as follows. In 1928, John von Neumann derived the "maximin" solution concept to solve zero-sum normal-form games [von Neumann, 1928]. Our proof of his minimax theorem is similar to the one in Luce and Raiffa [1957b]. In 1944, von Neumann together with Oskar Morgenstern authored what was to become the founding document of game theory [von Neumann and Morgenstern, 1944]; a second edition quickly followed in 1947. Among the many contributions of this work are the axiomatic foundations for "objective probabilities" and what became known as von Neumann–Morgenstern utility theory. The classical foundation of "subjective probabilities" is Savage [1954], but we do not cover those since they do not play a role in the book. A comprehensive overview of these foundational topics is provided by Kreps [1988], among others. Our own treatment of utility theory draws on Poole et al. [1997]; see also Russell and Norvig [2003].

But von Neumann and Morgenstern [1944] did much more; they introduced the normal-form game, the extensive form (to be discussed in Chapter 5), the concepts of pure and mixed strategies, as well as other notions central to game theory. Schelling [1960] was one of the first to show that interesting social interactions could usefully be modeled using game theory, for which he was recognized in 2005 with a Nobel Prize.

Shortly afterward John Nash introduced the concept of what would become known as the "Nash equilibrium" [Nash, 1950, 1951], without a doubt the most influential concept in game theory to this date. Indeed, Nash received a Nobel Prize in 1994 because of this work.[9] The proof in Nash [1950] uses Kakutani's fixed-point theorem; our proof of Theorem 3.3.22 follows Nash [1951]. Lemma 3.3.14 is due to Sperner [1928] and Theorem 3.3.17 is due to Brouwer [1912]; our proof of the latter follows Border [1985].

This work opened the floodgates to a series of refinements and alternative solution concepts which continues to this day. We covered several of these solution concepts. The literature on Pareto optimality and social optimization dates back to the early twentieth century, including seminal work by Pareto and Pigou, but perhaps was best established by Arrow in his seminal work on social choice [Arrow, 1970]. The minimax regret decision criterion was first proposed by Savage [1954], and further developed in Loomes and Sugden [1982] and Bell [1982]. Recent work from a computer science perspective includes Hyafil and Boutilier [2004], which also applies this criterion to the Bayesian games setting we introduce in Section 6.3. Iterated removal of dominated strategies, and the closely

9. John Nash was also the topic of the Oscar-winning 2001 movie *A Beautiful Mind*; however, the movie had little to do with his scientific contributions and indeed got the definition of Nash equilibrium wrong.

related rationalizability, enjoy a long history, though modern discussion of them is most firmly anchored in two independent and concurrent publications: Pearce [1984] and Bernheim [1984]. Correlated equilibria were introduced in Aumann [1974]; Myerson's quote is taken from Solan and Vohra [2002]. Trembling-hand perfection was introduced in Selten [1975]. An even stronger notion than (trembling-hand) perfect equilibrium is that of proper equilibrium [Myerson, 1978]. In Chapter 7 we discuss the concept of evolutionarily stable strategies [Maynard Smith and Price, 1973] and their connection to Nash equilibria. In addition to such single-equilibrium concepts, there are concepts that apply to sets of equilibria, not single ones. Of note are the notions of *stable equilibria* as originally defined in Kohlberg and Mertens [1986], and various later refinements such as *hyperstable sets* defined in Govindan and Wilson [2005a]. Good surveys of many of these concepts can be found in Hillas and Kohlberg [2002] and Govindan and Wilson [2005b].

stable equilibrium

hyperstable set

# 5

# Games with Sequential Actions: Reasoning and Computing with the Extensive Form

In Chapter 3 we assumed that a game is represented in normal form: effectively, as a big table. In some sense, this is reasonable. The normal form is conceptually straightforward, and most see it as fundamental. While many other representations exist to describe finite games, we will see in this chapter and in Chapter 6 that each of them has an "induced normal form": a corresponding normal-form representation that preserves game-theoretic properties such as Nash equilibria. Thus the results given in Chapter 3 hold for all finite games, no matter how they are represented; in that sense the normal-form representation is universal.

In this chapter we will look at extensive-form games, a finite representation that does not always assume that players act simultaneously. This representation is in general exponentially smaller than its induced normal form, and furthermore can be much more natural to reason about. While the Nash equilibria of an extensive-form game can be found through its induced normal form, computational benefit can be had by working with the extensive form directly. Furthermore, there are other solution concepts, such as subgame-perfect equilibrium (see Section 5.1.3), which explicitly refer to the sequence in which players act and which are therefore not meaningful when applied to normal-form games.

## 5.1 Perfect-information extensive-form games

The normal-form game representation does not incorporate any notion of sequence, or time, of the actions of the players. The *extensive* (or *tree) form* is an alternative representation that makes the temporal structure explicit. We start by discussing the special case of *perfect information* extensive-form games, and then move on to discuss the more general class of *imperfect-information* extensive-form games in Section 5.2. In both cases we will restrict the discussion to finite games, that is, to games represented as finite trees.

### 5.1.1 *Definition*

Informally speaking, a perfect-information game in extensive form (or, more simply, a perfect-information game) is a tree in the sense of graph theory, in which each node represents the choice of one of the players, each edge represents a possible action, and the leaves represent final outcomes over which each player has a utility function. Indeed, in certain circles (in particular, in artificial intelligence), these are known simply as game trees. Formally, we define them as follows.

Perfect-information game

**Definition 5.1.1 (Perfect-information game)** *A (finite) perfect-information game (in extensive form) is a tuple* $G = (N, A, H, Z, \chi, \rho, \sigma, u)$*, where:*

- $N$ *is a set of n players;*
- $A$ *is a (single) set of actions;*
- $H$ *is a set of nonterminal choice nodes;*
- $Z$ *is a set of terminal nodes, disjoint from* $H$*;*
- $\chi : H \mapsto 2^A$ *is the action function, which assigns to each choice node a set of possible actions;*
- $\rho : H \mapsto N$ *is the player function, which assigns to each nonterminal node a player* $i \in N$ *who chooses an action at that node;*
- $\sigma : H \times A \mapsto H \cup Z$ *is the successor function, which maps a choice node and an action to a new choice node or terminal node such that for all* $h_1, h_2 \in H$ *and* $a_1, a_2 \in A$*, if* $\sigma(h_1, a_1) = \sigma(h_2, a_2)$ *then* $h_1 = h_2$ *and* $a_1 = a_2$*; and*
- $u = (u_1, \ldots, u_n)$*, where* $u_i : Z \mapsto \mathbb{R}$ *is a real-valued utility function for player i on the terminal nodes* $Z$*.*

Since the choice nodes form a tree, we can unambiguously identify a node with its *history*, that is, the sequence of choices leading from the root node to it. We can also define the *descendants* of a node $h$, namely all the choice and terminal nodes in the subtree rooted at $h$.

An example of such a game is the *Sharing game*. Imagine a brother and sister following the following protocol for sharing two indivisible and identical presents from their parents. First the brother suggests a split, which can be one of three—he keeps both, she keeps both, or they each keep one. Then the sister chooses whether to accept or reject the split. If she accepts they each get their allocated present(s), and otherwise neither gets any gift. Assuming both siblings value the two presents equally and additively, the tree representation of this game is shown in Figure 5.1.

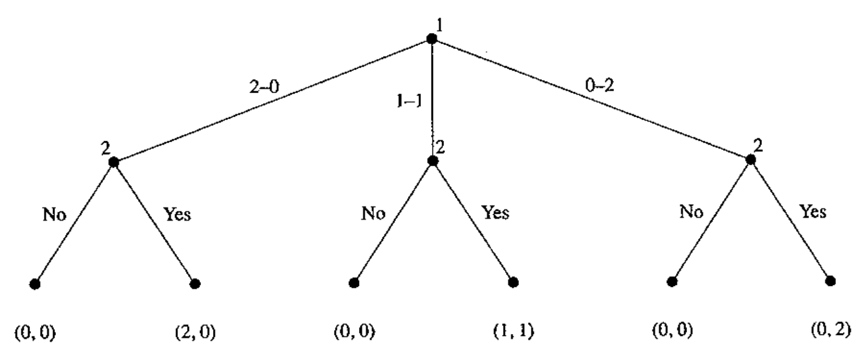


**Figure 5.1** The Sharing game.

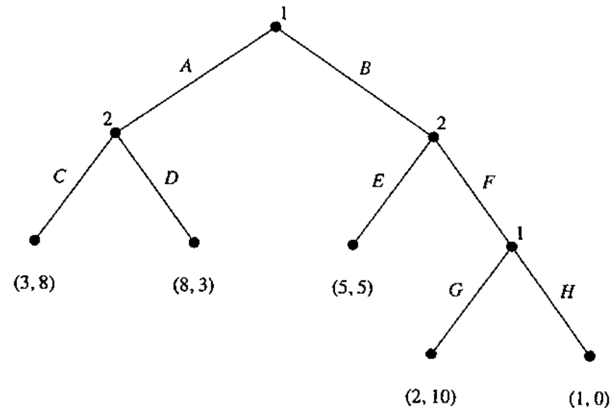


**Figure 5.2** A perfect-information game in extensive form.

## 5.1.2 *Strategies and equilibria*

A pure strategy for a player in a perfect-information game is a complete specification of which deterministic action to take at every node belonging to that player. A more formal definition follows.

**Definition 5.1.2 (Pure strategies)** *Let* $G = (N, A, H, Z, \chi, \rho, \sigma, u)$ *be a perfect-information extensive-form game. Then the pure strategies of player* $i$ *consist of the Cartesian product* $\prod_{h\in H,\rho(h)=i} \chi(h)$.

Notice that the definition contains a subtlety. An agent's strategy requires a decision at each choice node, regardless of whether or not it is possible to reach that node given the other choice nodes. In the Sharing game above the situation is straightforward—player 1 has three pure strategies, and player 2 has eight, as follows.

$$S_1 = \{2\text{–}0, 1\text{–}1, 0\text{–}2\}$$

$$S_2 = \{(yes, yes, yes), (yes, yes, no), (yes, no, yes), (yes, no, no), (no, yes, yes), (no, yes, no), (no, no, yes), (no, no, no)\}$$

But now consider the game shown in Figure 5.2.

In order to define a complete strategy for this game, each of the players must choose an action at each of his two choice nodes. Thus we can enumerate the pure strategies of the players as follows.

$$S_1 = \{(A, G), (A, H), (B, G), (B, H)\}$$

$$S_2 = \{(C, E), (C, F), (D, E), (D, F)\}$$

It is important to note that we have to include the strategies $(A, G)$ and $(A, H)$, even though once player 1 has chosen $A$ then his own $G$-versus-$H$ choice is moot.

| | $(C, E)$ | $(C, F)$ | $(D, E)$ | $(D, F)$ |
|---|---|---|---|---|
| $(A, G)$ | 3, 8 | 3, 8 | 8, 3 | 8, 3 |
| $(A, H)$ | 3, 8 | 3, 8 | 8, 3 | 8, 3 |
| $(B, G)$ | 5, 5 | 2, 10 | 5, 5 | 2, 10 |
| $(B, H)$ | 5, 5 | 1, 0 | 5, 5 | 1, 0 |

**Figure 5.3** The game from Figure 5.2 in normal form.

The definition of best response and Nash equilibria in this game are exactly as they are for normal-form games. Indeed, this example illustrates how every perfect-information game can be converted to an equivalent normal-form game. For example, the perfect-information game of Figure 5.2 can be converted into the normal-form image of the game, shown in Figure 5.3. Clearly, the strategy spaces of the two games are the same, as are the pure-strategy Nash equilibria. (Indeed, both the mixed strategies and the mixed-strategy Nash equilibria of the two games are also the same; however, we defer further discussion of mixed strategies until we consider imperfect-information games in Section 5.2.)

In this way, for every perfect-information game there exists a corresponding normal-form game. Note, however, that the temporal structure of the extensive-form representation can result in a certain redundancy within the normal form. For example, in Figure 5.3 there are 16 different outcomes, while in Figure 5.2 there are only 5 (the payoff (3, 8) occurs only once in Figure 5.2 but four times in Figure 5.3 etc.). One general lesson is that while this transformation can always be performed, it can result in an exponential blowup of the game representation. This is an important lesson, since the didactic examples of normal-form games are very small, wrongly suggesting that this form is more compact.

The normal form gets its revenge, however, since the reverse transformation—from the normal form to the perfect-information extensive form—does not always exist. Consider, for example, the Prisoner's Dilemma game from Figure 3.3. A little experimentation will convince the reader that there does not exist a perfect-information game that is equivalent in the sense of having the same strategy profiles and the same payoffs. Intuitively, the problem is that perfect-information extensive-form games cannot model simultaneity. The general characterization of the class of normal-form games for which there exist corresponding perfect-information games in extensive form is somewhat complex.

The reader will have noticed that we have so far concentrated on pure strategies and pure Nash equilibria in extensive-form games. There are two reasons for this, or perhaps one reason and one excuse. The reason is that mixed strategies introduce a new subtlety, and it is convenient to postpone discussion of

| | (C, E) | (C, F) | (D, E) | (D, F) |
|---|---|---|---|---|
| (A, G) | 3, 8 | 3, 8 | 8, 3 | 8, 3 |
| (A, H) | 3, 8 | 3, 8 | 8, 3 | 8, 3 |
| (B, G) | 5, 5 | 2, 10 | 5, 5 | 2, 10 |
| (B, H) | 5, 5 | 1, 0 | 5, 5 | 1, 0 |

**Figure 5.4** Equilibria of the game from Figure 5.2.

it. The excuse (which also allows the postponement, though not for long) is the following theorem.

**Theorem 5.1.3** *Every (finite) perfect-information game in extensive form has a pure-strategy Nash equilibrium.*

This is perhaps the earliest result in game theory, due to Zermelo in 1913 (see the historical notes at the end of the chapter). The intuition here should be clear; since players take turns, and everyone gets to see everything that happened thus far before making a move, it is never necessary to introduce randomness into action selection in order to find an equilibrium. We will see this plainly when we discuss *backward induction* below. Both this intuition and the theorem will cease to hold when we discuss more general classes of games such as imperfect-information games in extensive form. First, however, we discuss an important refinement of the concept of Nash equilibrium.

backward induction

### 5.1.3 *Subgame-perfect equilibrium*

As we have discussed, the notion of Nash equilibrium is as well defined in perfect-information games in extensive form as it is in the normal form. However, as the following example shows, the Nash equilibrium can be too weak a notion for the extensive form. Consider again the perfect-information extensive-form game shown in Figure 5.2. There are three pure-strategy Nash equilibria in this game: $\{(A, G), (C, F)\}$, $\{(A, H), (C, F)\}$, and $\{(B, H), (C, E)\}$. This can be determined by examining the normal form image of the game, as indicated in Figure 5.4.

However, examining the normal form image of an extensive-form game obscures the game's temporal nature. To illustrate a problem that can arise in certain equilibria of extensive-form games, in Figure 5.5 we contrast the equilibria $\{(A, G), (C, F)\}$ and $\{(B, H), (C, E)\}$ by drawing them on the extensive-form game tree.

First consider the equilibrium $\{(A, G), (C, F)\}$. If player 1 chooses $A$ then player 2 receives a higher payoff by choosing $C$ than by choosing $D$. If player 2

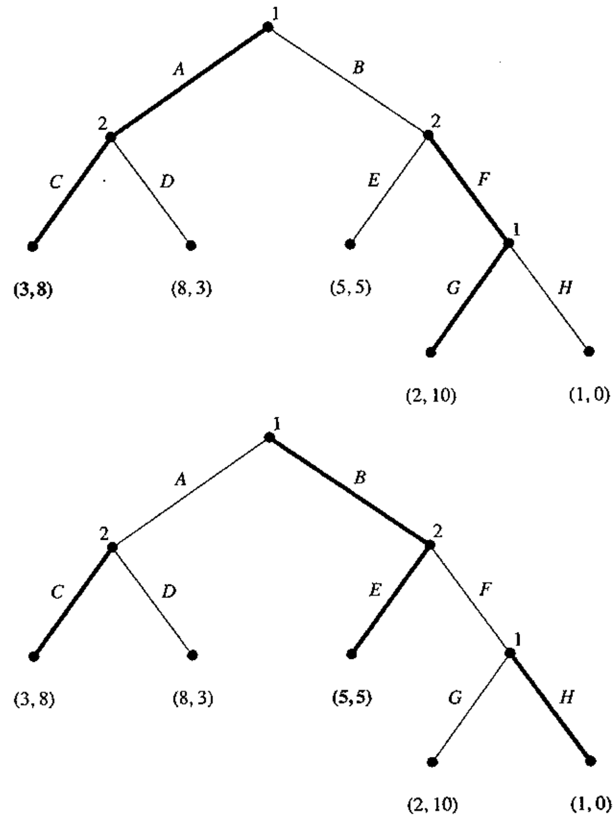


**Figure 5.5** Two out of the three equilibria of the game from Figure 5.2: $\{(A, G), (C, F)\}$ and $\{(B, H), (C, E)\}$. Bold edges indicate players' choices at each node.

played the strategy $(C, E)$ rather than $(C, F)$ then player 1 would prefer to play $B$ at the first node in the tree; as it is, player 1 gets a payoff of 3 by playing $A$ rather than a payoff of 2 by playing $B$. Hence we have an equilibrium.

The second equilibrium $\{(B, H), (C, E)\}$ is less intuitive. First, note that $\{(B, G), (C, E)\}$ is *not* an equilibrium: player 2's best response to $(B, G)$ is $(C, F)$. Thus, the only reason that player 2 chooses to play the action $E$ is that he knows that player 1 would play $H$ at his second decision node. This behavior by player 1 is called a *threat*: by committing to choose an action that is harmful to player 2 in his second decision node, player 1 can cause player 2 to avoid that part of the tree. (Note that player 1 benefits from making this threat: he gets a payoff of 5 instead of 2 by playing $(B, H)$ instead of $(B, G)$.) So far so good. The problem, however, is that player 2 may not consider player 1's threat to be credible: if player 1 did reach his final decision node, actually choosing $H$ over $G$ would also reduce player 1's own utility. If player 2 played $F$, would player 1 really follow through on his threat and play $H$, or would he relent and pick $G$ instead?

To formally capture the reason why the $\{(B, H), (C, E)\}$ equilibrium is unsatisfying, and to define an equilibrium refinement concept that does not suffer from this problem, we first define the notion of a subgame.

**Definition 5.1.4 (Subgame)** *Given a perfect-information extensive-form game $G$, the* subgame *of $G$ rooted at node $h$ is the restriction of $G$ to the descendants of $h$. The set of subgames of $G$ consists of all of subgames of $G$ rooted at some node in $G$.*

Now we can define the notion of a *subgame-perfect equilibrium*, a refinement of the Nash equilibrium in perfect-information games in extensive form, which eliminates those unwanted Nash equilibria.[1]

subgame-perfect equilibrium (SPE)

**Definition 5.1.5 (Subgame-perfect equilibrium)** *The* subgame-perfect equilibria *(SPE) of a game $G$ are all strategy profiles $s$ such that for any subgame $G'$ of $G$, the restriction of $s$ to $G'$ is a Nash equilibrium of $G'$.*

Since $G$ is its own subgame, every SPE is also a Nash equilibrium. Furthermore, although SPE is a stronger concept than Nash equilibrium (i.e., every SPE is a NE, but not every NE is a SPE) it is still the case that every perfect-information extensive-form game has at least one subgame-perfect equilibrium.

This definition rules out "noncredible threats" of the sort illustrated in the above example. In particular, note that the extensive-form game in Figure 5.2 has only one subgame-perfect equilibrium, $\{(A, G), (C, F)\}$. Neither of the other Nash equilibria is subgame perfect. Consider the subgame rooted at player 1's second choice node. The unique Nash equilibrium of this (trivial) game is for player 1 to play $G$. Thus the action $H$, the restriction of the strategies $(A, H)$ and $(B, H)$ to this subgame, is not optimal in this subgame, and cannot be part of a subgame-perfect equilibrium of the larger game.

### 5.1.4 *Computing equilibria: backward induction*

#### *n*-player, general-sum games: the backward induction algorithm

backward induction

Inherent in the concept of subgame-perfect equilibrium is the principle of *backward induction*. One identifies the equilibria in the "bottom-most" subgame trees, and assumes that those equilibria will be played as one backs up and considers increasingly larger trees. We can use this procedure to compute a sample Nash equilibrium. This is good news: not only are we guaranteed to find a subgame-perfect equilibrium (rather than possibly finding a Nash equilibrium that involves noncredible threats), but also this procedure is computationally simple. In particular, it can be implemented as a single depth-first traversal of the game tree and thus requires time linear in the size of the game representation. Recall in contrast that the best known methods for finding Nash equilibria of general games require time exponential in the size of the normal form; remember as well that the induced normal form of an extensive-form game is exponentially larger than the original representation.

The algorithm BACKWARDINDUCTION is described in Figure 5.6. The variable *util_at_child* is a vector denoting the utility for each player at the child node; $util\_at\_child_{\rho(h)}$ denotes the element of this vector corresponding to the utility for

1. Note that the word "perfect" is used in two different senses here.

```
function BackwardInduction (node h) returns u(h)
if h ∈ Z then
    return u(h)                                   // h is a terminal node
best_util ← −∞
forall a ∈ χ(h) do
    util_at_child ← BackwardInduction(σ(h, a))
    if util_at_child_ρ(h) > best_util_ρ(h) then
        best_util ← util_at_child
return best_util
```

**Figure 5.6** Procedure for finding the value of a sample (subgame-perfect) Nash equilibrium of a perfect-information extensive-form game.

player $\rho(h)$ (the player who gets to move at node $h$). Similarly, *best_util* is a vector giving utilities for each player.

Observe that this procedure does not return an equilibrium strategy for each of the $n$ players, but rather describes how to label each node with a vector of $n$ real numbers. This labeling can be seen as an extension of the game's utility function to the nonterminal nodes $H$. The players' equilibrium strategies follow straightforwardly from this extended utility function: every time a given player $i$ has the opportunity to act at a given node $h \in H$ (i.e., $\rho(h) = i$), that player will choose an action $a_i \in \chi(h)$ that solves $\arg\max_{a_i \in \chi(h)} u_i(\sigma(a_i, h))$. These strategies can also be returned by BackwardInduction given some extra bookkeeping.

While the procedure demonstrates that in principle a sample SPE is effectively computable, in practice many game trees are not enumerated in advance and are hence unavailable for backward induction. For example, the extensive-form representation of chess has around $10^{150}$ nodes, which is vastly too large to represent explicitly. For such games it is more common to discuss the size of the game tree in terms of the average branching factor $b$ (the average number of actions which are possible at each node) and a maximum depth $m$ (the maximum number of sequential actions). A procedure which requires time linear in the size of the representation thus expands $O(b^m)$ nodes. Unfortunately, we can do no better than this on arbitrary perfect-information games.

### Two-player, zero-sum games: minimax and alpha-beta pruning

We *can* make some computational headway in the widely applicable case of two-player, zero-sum games. We first note that BackwardInduction has another name in the two-player, zero-sum context: the *minimax algorithm*. Recall that in such games, only a single payoff number is required to characterize any outcome. Player 1 wants to maximize this number, while player 2 wants to minimize it. In this context BackwardInduction can be understood as propagating these single payoff numbers from the root of the tree up to the root. Each decision node for player 1 is labeled with the maximum of the labels of its child nodes (representing the fact that player 1 would choose the corresponding action), and each decision node for player 2 is labeled with the minimum of that node's

minimax algorithm

# 6

# Richer Representations: Beyond the Normal and Extensive Forms

In this chapter we will go beyond the normal and extensive forms by considering a variety of richer game representations. These further representations are important because the normal and extensive forms are not always suitable for modeling large or realistic game-theoretic settings.

First, we may be interested in games that are not finite and that therefore cannot be represented in normal or extensive form. For example, we may want to consider what happens when a simple normal-form game such as the Prisoner's Dilemma is repeated infinitely. We might want to consider a game played by an uncountably infinite set of agents. Or we may want to use an interval of the real numbers as each player's action space.[1]

Second, both of the representations we have studied so far presume that agents have perfect knowledge of everyone's payoffs. This seems like a poor model of many realistic situations, where, for example, agents might have private information that affects their own payoffs and other agents might have only probabilistic information about each others' private information. An elaboration like this can have a big impact, because one agent's actions can depend on what he knows about another agent's payoffs.

Finally, as the numbers of players and actions in a game grow—even if they remain finite—games can quickly become far too large to reason about or even to write down using the representations we have studied so far. Luckily, we are not usually interested in studying arbitrary strategic situations. The sorts of noncooperative settings that are most interesting in practice tend to involve highly structured payoffs. This can occur because of constraints imposed by the fact that the play of a game actually unfolds over time (e.g., because a large game actually corresponds to finitely repeated play of a small game). It can also occur because of the nature of the problem domain (e.g., while the world may involve many agents, the number of agents who are able to directly affect any given agent's payoff is small). If we understand the way in which agents' payoffs are structured, we can represent them much more compactly than we would be able to do using

1. We will explore the first example in detail in this chapter. A thorough treatment of infinite sets of players or action spaces is beyond the scope of this book; nevertheless, we will consider certain games with infinite sets of players in Section 6.4.4 and with infinite action spaces in Chapters 10 and 11.

the normal or extensive forms. Often, these compact representations also allow us to reason more efficiently about the games they describe (e.g., the computation of Nash equilibria can be provably faster, or pure-strategy Nash equilibria can be proved to always exist).

In this chapter we will present various different representations that address these limitations of the normal and extensive forms. In Section 6.1 we will begin by considering the special case of extensive-form games that are constructed by repeatedly playing a normal-form game and then we will extend our consideration to the case where the normal form is repeated infinitely. This will lead us to stochastic games in Section 6.2, which are like repeated games but do not require that the same normal-form game is played in each time step. In Section 6.3 we will consider structure of a different kind: instead of considering time, we will consider games involving uncertainty. Specifically, in Bayesian games agents face uncertainty—and hold private information—about the game's payoffs. Section 6.4 describes congestion games, which model situations in which agents contend for scarce resources. Finally, in Section 6.5 we will consider representations that are motivated primarily by compactness and by their usefulness for permitting efficient computation (e.g., of Nash equilibria). Such compact representations can extend any other existing representation, such as normal-form games, extensive-form games, or Bayesian games.

## 6.1 Repeated games

In repeated games, a given game (often thought of in normal form) is played multiple times by the same set of players. The game being repeated is called the *stage game*. For example, Figure 6.1 depicts two players playing the Prisoner's Dilemma exactly twice in a row.

stage game

| | *C* | *D* |
|---|---|---|
| *C* | −1, −1 | −4, 0 |
| *D* | 0, −4 | −3, −3 |

⇒

| | *C* | *D* |
|---|---|---|
| *C* | −1, −1 | −4, 0 |
| *D* | 0, −4 | −3, −3 |

**Figure 6.1** Twice-played Prisoner's Dilemma.

This representation of the repeated game, while intuitive, obscures some key factors. Do agents see what the other agents played earlier? Do they remember what they knew? And, while the utility of each stage game is specified, what is the utility of the entire repeated game?

We answer these questions in two steps. We first consider the case in which the game is repeated a finite and commonly-known number of times. Then we

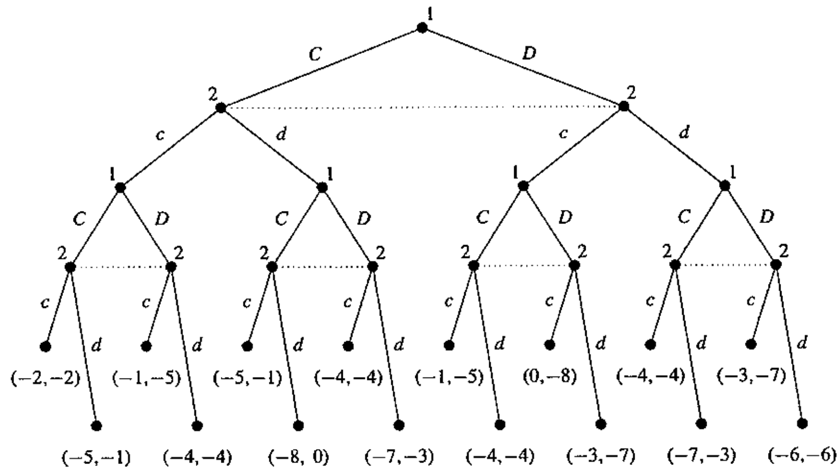


**Figure 6.2** Twice-played Prisoner's Dilemma in extensive form.

consider the case in which the game is repeated infinitely often, or a finite but unknown number of times.

## 6.1.1 *Finitely repeated games*

One way to completely disambiguate the semantics of a finitely repeated game is to specify it as an imperfect-information game in extensive form. Figure 6.2 describes the twice-played Prisoner's Dilemma game in extensive form. Note that it captures the assumption that at each iteration the players do not know what the other player is playing, but afterward they do. Also note that the payoff function of each agent is additive; that is, it is the sum of payoffs in the two-stage games.

The extensive form also makes it clear that the strategy space of the repeated game is much richer than the strategy space in the stage game. Certainly one strategy in the repeated game is to adopt the same strategy in each stage game; clearly, this memory less strategy, called a *stationary strategy*, is a behavioral strategy in the extensive-form representation of the game. But in general, the action (or mixture of actions) played at a stage game can depend on the history of play thus far. Since this fact plays a particularly important role in infinitely repeated games, we postpone further discussion of it to the next section. Indeed, in the finite, known repetition case, we encounter again the phenomenon of backward induction, which we first encountered when we introduced subgame-perfect equilibria. Recall that in the Centipede game, discussed in Section 5.1.3, the unique SPE was to go down and terminate the game at every node. Now consider a finitely repeated Prisoner's Dilemma game. Again, it can be argued, in the last round it is a dominant strategy to defect, no matter what happened so far. This is common knowledge, and no choice of action in the preceding rounds will impact the play in the last round. Thus in the second-to-last round too it is a dominant strategy to defect. Similarly, by induction, it can be argued that the only equilibrium in this case is to always defect. However, as in the case of the

stationary strategy

Centipede game, this argument is vulnerable to both empirical and theoretical criticisms.

### 6.1.2 *Infinitely repeated games*

When the infinitely repeated game is transformed into extensive form, the result is an infinite tree. So the payoffs cannot be attached to any terminal nodes, nor can they be defined as the sum of the payoffs in the stage games (which in general will be infinite). There are two common ways of defining a player's payoff in an infinitely repeated game to get around this problem. The first is the average payoff of the stage game in the limit.[2]

average reward

**Definition 6.1.1 (Average reward)** *Given an infinite sequence of payoffs $r_i^{(1)}$, $r_i^{(2)}, \ldots$ for player $i$, the* average reward *of $i$ is*

$$\lim_{k\to\infty} \frac{\sum_{j=1}^{k} r_i^{(j)}}{k}.$$

The *future discounted reward* to a player at a certain point of the game is the sum of his payoff in the immediate stage game, plus the sum of future rewards discounted by a constant factor. This is a recursive definition, since the future rewards again give a higher weight to early payoffs than to later ones.

future discounted reward

**Definition 6.1.2 (Discounted reward)** *Given an infinite sequence of payoffs $r_i^{(1)}, r_i^{(2)}, \ldots$ for player $i$, and a discount factor $\beta$ with $0 \le \beta \le 1$, the* future discounted reward *of $i$ is $\sum_{j=1}^{\infty} \beta^j r_i^{(j)}$.*

The discount factor can be interpreted in two ways. First, it can be taken to represent the fact that the agent cares more about his well-being in the near term than in the long term. Alternatively, it can be assumed that the agent cares about the future just as much as he cares about the present, but with some probability the game will be stopped any given round; $1 - \beta$ represents that probability. The analysis of the game is not affected by which perspective is adopted.

Tit-for-Tat (TfT)

Now let us consider strategy spaces in an infinitely repeated game. In particular, consider the infinitely repeated Prisoner's Dilemma game. As we discussed, there are many strategies other than stationary ones. One of the most famous is *Tit-for-Tat*. TfT is the strategy in which the player starts by cooperating and thereafter chooses in round $j + 1$ the action chosen by the other player in round $j$. Beside being both simple and easy to compute, this strategy is notoriously hard to beat; it was the winner in several repeated Prisoner's Dilemma competitions for computer programs.

trigger strategy

Since the space of strategies is so large, a natural question is whether we can characterize all the Nash equilibria of the repeated game. For example, if the discount factor is large enough, both players playing TfT is a Nash equilibrium. But there is an infinite number of others. For example, consider the *trigger strategy*. This is a draconian version of TfT; in the trigger strategy, a player starts

2. The observant reader will notice a potential difficulty in this definition, since the limit may not exist. One can extend the definition to cover these cases by using the lim sup operator in Definition 6.1.1 rather than lim.

by cooperating, but if ever the other player defects then the first defects forever. Again, for sufficiently large discount factor, the trigger strategy forms a Nash equilibrium not only with itself but also with TfT.

The folk theorem—so-called because it was part of the common lore before it was formally written down—helps us understand the space of all Nash equilibria of an infinitely repeated game, by answering a related question. It does not characterize the equilibrium strategy profiles, but rather the payoffs obtained in them. Roughly speaking, it states that in an infinitely repeated game the set of average rewards attainable in equilibrium are precisely those pairs attainable under mixed strategies in a single-stage game, with the constraint on the mixed strategies that each player's payoff is at least the amount he would receive if the other players adopted minmax strategies against him.

More formally, consider any $n$-player game $G = (N, A, u)$ and any payoff profile $r = (r_1, r_2, \dots, r_n)$. Let

$$v_i = \min_{s_{-i} \in S_{-i}} \max_{s_i \in S_i} u_i(s_{-i}, s_i).$$

In words, $v_i$ is player $i$'s minmax value: his utility when the other players play minmax strategies against him, and he plays his best response.

Before giving the theorem, we provide some more definitions.

**Definition 6.1.3 (Enforceable)** *A payoff profile $r = (r_1, r_2, \dots, r_n)$ is* enforceable *if $\forall i \in N$, $r_i \geq v_i$.*

**Definition 6.1.4 (Feasible)** *A payoff profile $r = (r_1, r_2, \dots, r_n)$ is* feasible *if there exist rational, nonnegative values $\alpha_a$ such that for all $i$, we can express $r_i$ as $\sum_{a \in A} \alpha_a u_i(a)$, with $\sum_{a \in A} \alpha_a = 1$.*

In other words, a payoff profile is feasible if it is a convex, rational combination of the outcomes in $G$.

folk theorem

**Theorem 6.1.5 (Folk Theorem)** *Consider any $n$-player normal-form game $G$ and any payoff profile $r = (r_1, r_2, \dots, r_n)$.*

1. *If $r$ is the payoff profile for any Nash equilibrium $s$ of the infinitely repeated $G$ with average rewards, then for each player $i$, $r_i$ is enforceable.*
2. *If $r$ is both feasible and enforceable, then $r$ is the payoff profile for some Nash equilibrium of the infinitely repeated $G$ with average rewards.*

This proof is both instructive and intuitive. The first part uses the definition of minmax and best response to show that an agent can never receive less than his minmax value in any equilibrium. The second part shows how to construct an equilibrium that yields each agent the average payoffs given in any feasible and enforceable payoff profile $r$. This equilibrium has the agents cycle in perfect lock-step through a sequence of game outcomes that achieve the desired average payoffs. If any agent deviates, the others punish him forever by playing their minmax strategies against him.

**Proof. Part 1:** Suppose $r$ is not enforceable, that is, $r_i < v_i$ for some $i$. Then consider an alternative strategy for $i$: playing $BR(s_{-i}(h))$, where $s_{-i}(h)$ is the equilibrium strategy of other players given the current history $h$ and $BR(s_{-i}(h))$ is a function that returns a best response for $i$ to a given strategy profile $s_{-i}$ in the (unrepeated) stage game $G$. By definition of a minmax strategy, player $i$ receives a payoff of at least $v_i$ in every stage game if he plays $BR(s_{-i}(h))$, and so $i$'s average reward is also at least $v_i$. Thus, if $r_i < v_i$ then $s$ cannot be a Nash equilibrium.

**Part 2:** Since $r$ is a feasible enforceable payoff profile, we can write it as $r_i = \sum_{a \in A}(\frac{\beta_a}{\gamma})u_i(a)$, where $\beta_a$ and $\gamma$ are nonnegative integers. (Recall that $\alpha_a$ were required to be rational. So we can take $\gamma$ to be their common denominator.) Since the combination was convex, we have $\gamma = \sum_{a \in A} \beta_a$.

We are going to construct a strategy profile that will cycle through all outcomes $a \in A$ of $G$ with cycles of length $\gamma$, each cycle repeating action $a$ exactly $\beta_a$ times. Let $(a^t)$ be such a sequence of outcomes. Let us define a strategy $s_i$ of player $i$ to be a trigger version of playing $(a^t)$: if nobody deviates, then $s_i$ plays $a_i^t$ in period $t$. However, if there was a period $t'$ in which some player $j \neq i$ deviated, then $s_i$ will play $(p_{-j})_i$, where $(p_{-j})$ is a solution to the minimization problem in the definition of $v_j$.

First observe that if everybody plays according to $s_i$, then, by construction, player $i$ receives average payoff of $r_i$ (look at averages over periods of length $\gamma$). Second, this strategy profile is a Nash equilibrium. Suppose everybody plays according to $s_i$, and player $j$ deviates at some point. Then, forever after, player $j$ will receive his min max payoff $v_j \leq r_j$, rendering the deviation unprofitable. ■

The reader might wonder why this proof appeals to $i$'s minmax value rather than his maxmin value. First, notice that the trigger strategies in Part 2 of the proof use minmax strategies to punish agent $i$. This makes sense because even in cases where $i$'s minmax value is strictly greater than his maxmin value,[3] $i$'s minmax value is the smallest amount that the other agents can guarantee that $i$ will receive. When $i$ best responds to a minmax strategy played against him by $-i$, he receives exactly his minmax value; this is the deviation considered in Part 1.

Theorem 6.1.5 is actually an instance of a large family of folk theorems. As stated, Theorem 6.1.5 is restricted to infinitely repeated games, to average reward, to the Nash equilibrium, and to games of complete information. However, there are folk theorems that hold for other versions of each of these conditions, as well as other conditions not mentioned here. In particular, there are folk theorems for infinitely repeated games with discounted reward (for a large enough discount factor), for finitely repeated games, for subgame-perfect equilibria (i.e., where agents only administer finite punishments to deviators), and for games of incomplete information. We do not review them here, but the message of each of them

3. This can happen in games with more than two players, as discussed in Section 3.4.1.

| | C | D |
|---|---|---|
| C | 3, 3 | 0, 4 |
| D | 4, 0 | 1, 1 |

**Figure 6.3** Prisoner's Dilemma game.

is fundamentally the same: the payoffs in the equilibria of a repeated game are essentially constrained only by enforceability and feasibility.

### 6.1.3 *"Bounded rationality": repeated games played by automata*

Until now we have assumed that players can engage in arbitrarily deep reasoning and mutual modeling, regardless of their complexity. In particular, consider the fact that we have tended to rely on equilibrium concepts as predictions of—or prescriptions for—behavior. Even in the relatively uncontroversial case of two-player zero-sum games, this is a questionable stance in practice; otherwise, for example, there would be no point in chess competitions. While we will continue to make this questionable assumption in much of the remainder of the book, we pause here to revisit it. We ask what happens when agents are not perfectly rational expected-utility maximizers. In particular, we ask what happens when we impose specific computational limitations on them.

Consider (yet again) an instance of the Prisoner's Dilemma, which is reproduced in Figure 6.3. In the finitely repeated version of this game, we know that each player's dominant strategy (and thus the only Nash equilibrium) is to choose the strategy $D$ in each iteration of the game. In reality, when people actually play the game, we typically observe a significant amount of cooperation, especially in the earlier iterations of the game. While much of game theory is open to the criticism that it does not match well with human behavior, this is a particularly stark example of this divergence. What models might explain this fact?

One early proposal in the literature is based on the notion of an $\epsilon$-equilibrium, defined in Section 3.4.7. Recall that this is a strategy profile in which no agent can gain more than $\epsilon$ by changing his strategy; a Nash equilibrium is thus the special case of a 0-equilibrium. This equilibrium concept is motivated by the idea that agents' rationality may be bounded in the sense that they are willing to settle for payoffs that are slightly below their best response payoffs. In the finitely repeated Prisoner's Dilemma game, as the number of repetitions increases, the corresponding sets of $\epsilon$-equilibria include outcomes with longer and longer sequences of the "cooperate" strategy.

Various other models of bounded rationality exist, but we will focus on what has proved to be the richest source of results so far, namely, restricting agents' strategies to those implemented by automata of the sort investigated in computer science.

games, an alternative is to use an algorithm developed by Shapley that is related to value iteration, a commonly-used method for solving MDPs (see Appendix C).

Moving on to the average reward case, we have to impose more restrictions in order to use a linear program than we did for the discounted reward case. Specifically, for the class of two-player, general-sum, average-reward stochastic games, the single-controller assumption no longer suffices—we also need the game to be zero sum.

Even when we cannot use a linear program, irreducibility allows us to use an algorithm that is guaranteed to converge. This algorithm is a combination of policy iteration (another method used for solving MDPs) and successive approximation.

## 6.3 Bayesian games

All of the game forms discussed so far assumed that all players know what game is being played. Specifically, the number of players, the actions available to each player, and the payoff associated with each action vector have all been assumed to be common knowledge among the players. Note that this is true even of imperfect-information games; the actual moves of agents are not common knowledge, but the game itself is. In contrast, *Bayesian games*, or games of incomplete information, allow us to represent players' uncertainties about the very game being played.[4] This uncertainty is represented as a probability distribution over a set of possible games. We make two assumptions.

Bayesian game

1. All possible games have the same number of agents and the same strategy space for each agent; they differ only in their payoffs.
2. The beliefs of the different agents are posteriors, obtained by conditioning a common prior on individual private signals.

The second assumption is substantive, and we return to it shortly. The first is not particularly restrictive, although at first it might seem to be. One can imagine many other potential types of uncertainty that players might have about the game—how many players are involved, what actions are available to each player, and perhaps other aspects of the situation. It might seem that we have severely limited the discussion by ruling these out. However, it turns out that these other types of uncertainty can be reduced to uncertainty only about payoffs via problem reformulation.

For example, imagine that we want to model a situation in which one player is uncertain about the number of actions available to the other players. We can reduce this uncertainty to uncertainty about payoffs by padding the game with irrelevant actions. For example, consider the following two-player game, in which the row player does not know whether his opponent has only the two strategies $L$ and $R$ or also the third one $C$:

4. It is easy to confuse the term "incomplete information" with "imperfect information"; don't...

| | $L$ | $R$ |
|---|---|---|
| $U$ | 1, 1 | 1, 3 |
| $D$ | 0, 5 | 1, 13 |

| | $L$ | $C$ | $R$ |
|---|---|---|---|
| $U$ | 1, 1 | 0, 2 | 1, 3 |
| $D$ | 0, 5 | 2, 8 | 1, 13 |

Now consider replacing the leftmost, smaller game by a padded version, in which we add a new $C$ column.

| | $L$ | $C$ | $R$ |
|---|---|---|---|
| $U$ | 1, 1 | 0, −100 | 1, 3 |
| $D$ | 0, 5 | 2, −100 | 1, 13 |

Clearly the newly added column is dominated by the others and will not participate in any Nash equilibrium (or any other reasonable solution concept). Indeed, there is an isomorphism between Nash equilibria of the original game and the padded one. Thus the uncertainty about the strategy space can be reduced to uncertainty about payoffs.

Using similar tactics, it can be shown that it is also possible to reduce uncertainty about other aspects of the game to uncertainty about payoffs only. This is not a mathematical claim, since we have given no mathematical characterization of all the possible forms of uncertainty, but it is the case that such reductions have been shown for all the common forms of uncertainty.

common-prior assumption

The second assumption about Bayesian games is the *common-prior assumption*, addressed in more detail in our discussion of multiagent probabilities and KP-structures in Chapter 13. As discussed there, a Bayesian game thus defines not only the uncertainties of agents about the game being played, but also their beliefs about the beliefs of other agents about the game being played, and indeed an entire infinite hierarchy of nested beliefs (the so-called epistemic type space). As also discussed in Chapter 13, the common-prior assumption is a substantive assumption that limits the scope of applicability. We nonetheless make this assumption since it allows us to formulate the main ideas in Bayesian games, and without the assumption the subject matter becomes much more involved than is appropriate for this text. Indeed, most (but not all) work in game theory makes this assumption.

### 6.3.1 *Definition*

There are several ways of presenting Bayesian games; we will offer three different definitions. All three are equivalent, modulo some subtleties that lie outside the

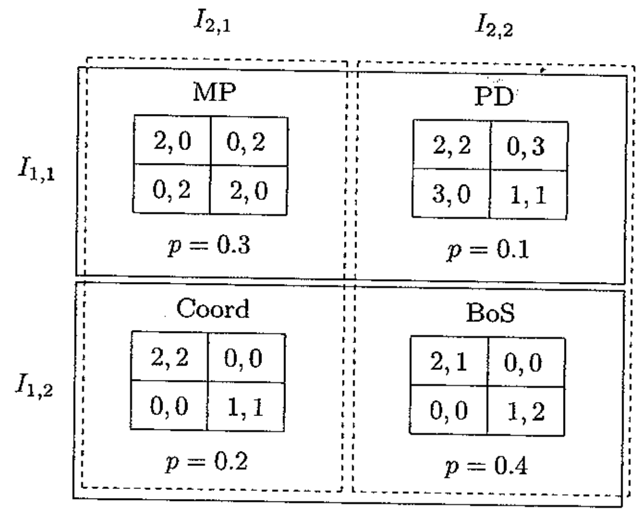


**Figure 6.7** A Bayesian game.

scope of this book. We include all three since each formulation is useful in different settings and offers different intuition about the underlying structure of this family of games.

### Information sets

First, we present a definition that is based on information sets. Under this definition, a Bayesian game consists of a set of games that differ only in their payoffs, a common prior defined over them, and a partition structure over the games for each agent.[5]

Bayesian game

**Definition 6.3.1 (Bayesian game: information sets)** *A* Bayesian game *is a tuple $(N, G, P, I)$ where:*

- *$N$ is a set of agents;*
- *$G$ is a set of games with $N$ agents each such that if $g, g' \in G$ then for each agent $i \in N$ the strategy space in $g$ is identical to the strategy space in $g'$;*
- *$P \in \Pi(G)$ is a common prior over games, where $\Pi(G)$ is the set of all probability distributions over $G$; and*
- *$I = (I_1, ..., I_N)$ is a tuple of partitions of $G$, one for each agent.*

Figure 6.7 gives an example of a Bayesian game. It consists of four $2 \times 2$ games (Matching Pennies, Prisoner's Dilemma, Coordination and Battle of the Sexes), and each agent's partition consists of two equivalence classes.

### Extensive form with chance moves

A second way of capturing the common prior is to hypothesize a special agent called Nature who makes probabilistic choices. While we could have Nature's

5. This combination of a common prior and a set of partitions over states of the world turns out to correspond to a KP-structure, defined in Chapter 13.

choice be interspersed arbitrarily with the agents' moves, without loss of generality we assume that Nature makes all its choices at the outset. Nature does not have a utility function (or, alternatively, can be viewed as having a constant one), and has the unique strategy of randomizing in a commonly known way. The agents receive individual signals about Nature's choice, and these are captured by their information sets in a standard way. The agents have no additional information; in particular, the information sets capture the fact that agents make their choices without knowing the choices of others. Thus, we have reduced games of incomplete information to games of imperfect information, albeit ones with chance moves. These chance moves of Nature require minor adjustments of existing definitions, replacing payoffs by their expectations given Nature's moves.[6]

For example, the Bayesian game of Figure 6.7 can be represented in extensive form as depicted in Figure 6.8.

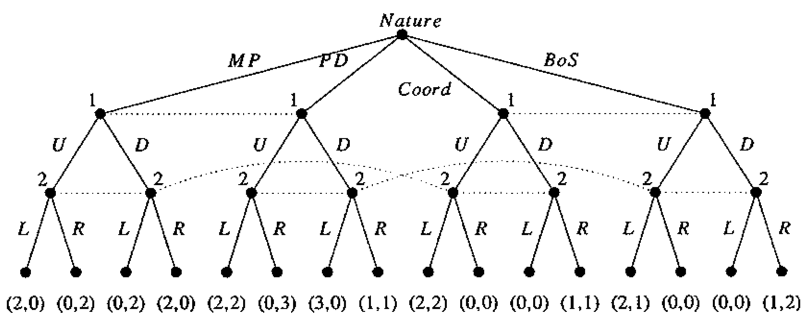


**Figure 6.8** The Bayesian game from Figure 6.7 in extensive form.

Although this second definition of Bayesian games can be initially more intuitive than our first definition, it can also be more cumbersome to work with. This is because we use an extensive-form representation in a setting where players are unable to observe each others' moves. (Indeed, for the same reason we do not routinely use extensive-form games of imperfect information to model simultaneous interactions such as the Prisoner's Dilemma, though we could do so if we wished.) For this reason, we will not make further use of this definition. We close by noting one advantage that it does have, however: it extends very naturally to Bayesian games in which players move sequentially and do (at least sometimes) learn about previous players' moves.

## Epistemic types

Recall that a game may be defined by a set of players, actions, and utility functions. In our first definition agents are uncertain about which game they are playing; however, each possible game has the same sets of actions and players, and so agents are really only uncertain about the game's utility function.

6. Note that the special structure of this extensive-form game means that we do not have to agonize over the refinements of Nash equilibrium; since agents have no information about prior choices made other than by Nature, all Nash equilibria are also sequential equilibria.

| $a_1$ | $a_2$ | $\theta_1$ | $\theta_2$ | $u_1$ | $u_2$ |
|---|---|---|---|---|---|
| U | L | $\theta_{1,1}$ | $\theta_{2,1}$ | 2 | 0 |
| U | L | $\theta_{1,1}$ | $\theta_{2,2}$ | 2 | 2 |
| U | L | $\theta_{1,2}$ | $\theta_{2,1}$ | 2 | 2 |
| U | L | $\theta_{1,2}$ | $\theta_{2,2}$ | 2 | 1 |
| U | R | $\theta_{1,1}$ | $\theta_{2,1}$ | 0 | 2 |
| U | R | $\theta_{1,1}$ | $\theta_{2,2}$ | 0 | 3 |
| U | R | $\theta_{1,2}$ | $\theta_{2,1}$ | 0 | 0 |
| U | R | $\theta_{1,2}$ | $\theta_{2,2}$ | 0 | 0 |

| $a_1$ | $a_2$ | $\theta_1$ | $\theta_2$ | $u_1$ | $u_2$ |
|---|---|---|---|---|---|
| D | L | $\theta_{1,1}$ | $\theta_{2,1}$ | 0 | 2 |
| D | L | $\theta_{1,1}$ | $\theta_{2,2}$ | 3 | 0 |
| D | L | $\theta_{1,2}$ | $\theta_{2,1}$ | 0 | 0 |
| D | L | $\theta_{1,2}$ | $\theta_{2,2}$ | 0 | 0 |
| D | R | $\theta_{1,1}$ | $\theta_{2,1}$ | 2 | 0 |
| D | R | $\theta_{1,1}$ | $\theta_{2,2}$ | 1 | 1 |
| D | R | $\theta_{1,2}$ | $\theta_{2,1}$ | 1 | 1 |
| D | R | $\theta_{1,2}$ | $\theta_{2,2}$ | 1 | 2 |

**Figure 6.9** Utility functions $u_1$ and $u_2$ for the Bayesian game from Figure 6.7.

epistemic type

Our third definition uses the notion of an *epistemic type*, or simply a *type* as a way of defining uncertainty directly over a game's utility function.

Bayesian game

**Definition 6.3.2 (Bayesian game: types)** *A* Bayesian game *is a tuple* $(N, A, \Theta, p, u)$ *where:*

- $N$ *is a set of agents;*
- $A = A_1 \times \cdots \times A_n$, *where* $A_i$ *is the set of actions available to player* $i$;
- $\Theta = \Theta_1 \times \ldots \times \Theta_n$, *where* $\Theta_i$ *is the type space of player* $i$;
- $p : \Theta \mapsto [0, 1]$ *is a common prior over types; and*
- $u = (u_1, \ldots, u_n)$, *where* $u_i : A \times \Theta \mapsto \mathbb{R}$ *is the utility function for player* $i$.

The assumption is that all of the above is common knowledge among the players, and that each agent knows his own type. This definition can seem mysterious, because the notion of type can be rather opaque. In general, the type of agent encapsulates all the information possessed by the agent that is not common knowledge. This is often quite simple (e.g., the agent's knowledge of his private payoff function), but can also include his beliefs about other agents' payoffs, about their beliefs about his own payoff, and any other higher-order beliefs.

We can get further insight into the notion of a type by relating it to the formulation at the beginning of this section. Consider again the Bayesian game in Figure 6.7. For each of the agents we have two types, corresponding to his two information sets. Denote player 1's actions as U and D, player 2's actions as L and R. Call the types of the first agent $\theta_{1,1}$ and $\theta_{1,2}$, and those of the second agent $\theta_{2,1}$ and $\theta_{2,2}$. The joint distribution on these types is as follows: $p(\theta_{1,1}, \theta_{2,1}) = .3$, $p(\theta_{1,1}, \theta_{2,2}) = .1$, $p(\theta_{1,2}, \theta_{2,1}) = .2$, $p(\theta_{1,2}, \theta_{2,2}) = .4$. The conditional probabilities for the first player are $p(\theta_{2,1} \mid \theta_{1,1}) = 3/4$, $p(\theta_{2,2} \mid \theta_{1,1}) = 1/4$, $p(\theta_{2,1} \mid \theta_{1,2}) = 1/3$, and $p(\theta_{2,2} \mid \theta_{1,2}) = 2/3$. Both players' utility functions are given in Figure 6.9.

### 6.3.2 *Strategies and equilibria*

Now that we have defined Bayesian games, we must explain how to reason about them. We will do this using the epistemic type definition, because that is the definition most commonly used in mechanism design (discussed in Chapter 10), one of the main applications of Bayesian games. All of the concepts defined below can also be expressed in terms of the first two Bayesian game definitions as well.

The first task is to define an agent's strategy space in a Bayesian game. Recall that in an imperfect-information extensive-form game a pure strategy is a mapping from information sets to actions. The definition is similar in Bayesian games: a pure strategy $\alpha_i : \Theta_i \mapsto A_i$ is a mapping from every type agent $i$ could have to the action he would play if he had that type. We can then define mixed strategies in the natural way as probability distributions over pure strategies. As before, we denote a mixed strategy for $i$ as $s_i \in S_i$, where $S_i$ is the set of all $i$'s mixed strategies. Furthermore, we use the notation $s_j(a_j|\theta_j)$ to denote the probability under mixed strategy $s_j$ that agent $j$ plays action $a_j$, given that $j$'s type is $\theta_j$.

Next, since we have defined an environment with multiple sources of uncertainty, we will pause to reconsider the definition of an agent's expected utility. In a Bayesian game setting, there are three meaningful notions of expected utility: *ex post*, *ex interim* and *ex ante*. The first is computed based on all agents' actual types, the second considers the setting in which an agent knows his own type but not the types of the other agents, and in the third case the agent does not know anybody's type.

*ex post* expected utility

**Definition 6.3.3 (*Ex post* expected utility)** *Agent $i$'s* ex post *expected utility in a Bayesian game $(N, A, \Theta, p, u)$, where the agents' strategies are given by $s$ and the agent' types are given by $\theta$, is defined as*

$$EU_i(s, \theta) = \sum_{a \in A} \left( \prod_{j \in N} s_j(a_j|\theta_j) \right) u_i(a, \theta). \tag{6.1}$$

In this case, the only uncertainty concerns the other agents' mixed strategies, since agent $i$'s *ex post* expected utility is computed based on the other agents' actual types. Of course, in a Bayesian game no agent *will* know the others' types; while that does not prevent us from offering the definition given, it might make the reader question its usefulness. We will see that this notion of expected utility is useful both for defining the other two and also for defining a specialized equilibrium concept.

**Definition 6.3.4 (*Ex interim* expected utility)** *Agent $i$'s* ex interim *expected utility in a Bayesian game $(N, A, \Theta, p, u)$, where $i$'s type is $\theta_i$ and where the agents' strategies are given by the mixed-strategy profile $s$, is defined as*

$$EU_i(s, \theta_i) = \sum_{\theta_{-i} \in \Theta_{-i}} p(\theta_{-i}|\theta_i) \sum_{a \in A} \left( \prod_{j \in N} s_j(a_j|\theta_j) \right) u_i(a, \theta_{-i}, \theta_i), \tag{6.2}$$

*or equivalently as*

$$EU_i(s, \theta_i) = \sum_{\theta_{-i} \in \Theta_{-i}} p(\theta_{-i}|\theta_i) EU_i(s, (\theta_i, \theta_{-i})). \tag{6.3}$$

Thus, $i$ must consider every assignment of types to the other agents $\theta_{-i}$ and every pure action profile $a$ in order to evaluate his utility function $u_i(a, \theta_i, \theta_{-i})$. He must weight this utility value by two amounts: the probability that the other players' types would be $\theta_{-i}$ given that his own type is $\theta_i$, and the probability that the pure action profile $a$ would be realized given all players' mixed strategies and types. (Observe that agents' types may be correlated.) Because uncertainty over mixed strategies was already handled in the *ex post* case, we can also write *ex interim* expected utility as a weighted sum of $EU_i(s, \theta)$ terms.

Finally, there is the *ex ante* case, where we compute $i$'s expected utility under the joint mixed strategy $s$ without observing any agents' types.

*ex ante* expected utility

**Definition 6.3.5 (*Ex ante* expected utility)** *Agent i's* ex ante *expected utility in a Bayesian game* $(N, A, \Theta, p, u)$*, where the agents' strategies are given by the mixed-strategy profile s, is defined as*

$$EU_i(s) = \sum_{\theta \in \Theta} p(\theta) \sum_{a \in A} \left( \prod_{j \in N} s_j(a_j | \theta_j) \right) u_i(a, \theta), \tag{6.4}$$

*or equivalently as*

$$EU_i(s) = \sum_{\theta \in \Theta} p(\theta) EU_i(s, \theta), \tag{6.5}$$

*or again equivalently as*

$$EU_i(s) = \sum_{\theta_i \in \Theta_i} p(\theta_i) EU_i(s, \theta_i). \tag{6.6}$$

Next, we define best response.

best response in a Bayesian game

**Definition 6.3.6 (Best response in a Bayesian game)** *The set of agent i's* best responses *to mixed-strategy profile* $s_{-i}$ *are given by*

$$BR_i(s_{-i}) = \arg\max_{s'_i \in S_i} EU_i(s'_i, s_{-i}). \tag{6.7}$$

Note that $BR_i$ is a set because there may be many strategies for $i$ that yield the same expected utility. It may seem odd that $BR$ is calculated based on $i$'s *ex ante* expected utility. However, write $EU_i(s)$ as $\sum_{\theta_i \in \Theta_i} p(\theta_i) EU_i(s, \theta_i)$ and observe that $EU_i(s'_i, s_{-i}, \theta_i)$ does not depend on strategies that $i$ would play if his type were not $\theta_i$. Thus, we are in fact performing independent maximization of $i$'s *ex interim* expected utilities conditioned on each type that he could have. Intuitively speaking, if a certain action is best after the signal is received, it is also the best conditional plan devised ahead of time for what to do should that signal be received.

We are now able to define the Bayes–Nash equilibrium.

Bayes–Nash equilibrium

**Definition 6.3.7 (Bayes–Nash equilibrium)** *A* Bayes–Nash equilibrium *is a mixed-strategy profile s that satisfies* $\forall i \; s_i \in BR_i(s_{-i})$.

This is exactly the definition we gave for the Nash equilibrium in Definition 3.3.4: each agent plays a best response to the strategies of the other players. The difference from Nash equilibrium, of course, is that the definition of Bayes–Nash equilibrium is built on top of the Bayesian game definitions of best response and expected utility. Observe that we would not be able to define equilibrium in this way if an agent's strategies were not defined for every possible type. In order for a given agent $i$ to play a best response to the other agents $-i$, $i$ must know what strategy each agent would play for each of his possible types. Without this information, it would be impossible to evaluate the term $EU_i(s_i', s_{-i})$ in Equation (6.7).

### 6.3.3 *Computing equilibria*

Despite its similarity to the Nash equilibrium, the Bayes–Nash equilibrium may seem more conceptually complicated. However, as we did with extensive-form games, we can construct a normal-form representation that corresponds to a given Bayesian game.

As with games in extensive form, the induced normal form for Bayesian games has an action for every pure strategy. That is, the actions for an agent $i$ are the distinct mappings from $\Theta_i$ to $A_i$. Each agent $i$'s payoff given a pure-strategy profile $s$ is his *ex ante* expected utility under $s$. Then, as it turns out, the Bayes–Nash equilibria of a Bayesian game are precisely the Nash equilibria of its induced normal form. This fact allows us to note that Nash's theorem applies directly to Bayesian games, and hence that Bayes–Nash equilibria always exist.

An example will help. Consider the Bayesian game from Figure 6.9. Note that in this game each agent has four possible pure strategies (two types and two actions). Then player 1's four strategies in the Bayesian game can be labeled $UU$, $UD$, $DU$, and $DD$: $UU$ means that 1 chooses $U$ regardless of his type, $UD$ that he chooses $U$ when he has type $\theta_{1,1}$ and $D$ when he has type $\theta_{1,2}$, and so forth. Similarly, we can denote the strategies of player 2 in the Bayesian game by $RR$, $RL$, $LR$, and $LL$.

We now define a $4 \times 4$ normal-form game in which these are the four strategies of the two agents, and the payoffs are the expected payoffs in the individual games, given the agents' common prior beliefs. For example, player 2's *ex ante* expected utility under the strategy profile $(UU, LL)$ is calculated as follows:

$$
\begin{aligned}
&u_2(UU, LL)\\
&\quad = \sum_{\theta \in \Theta} p(\theta) u_2(U, L, \theta)\\
&\quad = p(\theta_{1,1}, \theta_{2,1}) u_2(U, L, \theta_{1,1}, \theta_{2,1}) + p(\theta_{1,1}, \theta_{2,2}) u_2(U, L, \theta_{1,1}, \theta_{2,2})\\
&\qquad + p(\theta_{1,2}, \theta_{2,1}) u_2(U, L, \theta_{1,2}, \theta_{2,1}) + p(\theta_{1,2}, \theta_{2,2}) u_2(U, L, \theta_{1,2}, \theta_{2,2})\\
&\quad = 0.3(0) + 0.1(2) + 0.2(2) + 0.4(1) = 1.
\end{aligned}
$$

Continuing in this manner, the complete payoff matrix can be constructed as shown in Figure 6.10.

| | $LL$ | $LR$ | $RL$ | $RR$ |
|---|---|---|---|---|
| $UU$ | 2, 1 | 1, 0.7 | 1, 1.2 | 0, 0.9 |
| $UD$ | 0.8, 0.2 | 1, 1.1 | 0.4, 1 | 0.6, 1.9 |
| $DU$ | 1.5, 1.4 | 0.5, 1.1 | 1.7, 0.4 | 0.7, 0.1 |
| $DD$ | 0.3, 0.6 | 0.5, 1.5 | 1.1, 0.2 | 1.3, 1.1 |

**Figure 6.10** Induced normal form of the game from Figure 6.9.

Now the game may be analyzed straightforwardly. For example, we can determine that player 1's best response to $RL$ is $DU$.

Given a particular signal, the agent can compute the posterior probabilities and recompute the expected utility of any given strategy vector. Thus in the previous example once the row agent gets the signal $\theta_{1,1}$ he can update the expected payoffs and compute the new game shown in Figure 6.11.

| | $LL$ | $LR$ | $RL$ | $RR$ |
|---|---|---|---|---|
| $UU$ | 2, 0.5 | 1.5, 0.75 | 0.5, 2 | 0, 2.25 |
| $UD$ | 2, 0.5 | 1.5, 0.75 | 0.5, 2 | 0, 2.25 |
| $DU$ | 0.75, 1.5 | 0.25, 1.75 | 2.25, 0 | 1.75, 0.25 |
| $DD$ | 0.75, 1.5 | 0.25, 1.75 | 2.25, 0 | 1.75, 0.25 |

**Figure 6.11** *Ex interim* induced normal-form game, where player 1 observes type $\theta_{1,1}$.

Note that for the row player, $DU$ is still a best response to $RL$; what has changed is how much better it is compared to the other three strategies. In particular, the row player's payoffs are now independent of his choice of which action to take upon observing type $\theta_{1,2}$; in effect, conditional on observing type $\theta_{1,1}$ the player needs only to select a single action $U$ or $D$. (Thus, we could have written the *ex interim* induced normal form in Figure 6.11 as a table with four columns but only two rows.)

Although we can use this matrix to find best responses for player 1, it turns out to be meaningless to analyze the Nash equilibria in this payoff matrix. This is because these expected payoffs are not common knowledge; if the column player were to condition on his signal, he would arrive at a different set of numbers (though, again, for him best responses would be preserved). Ironically, it is only in the induced normal form, in which the payoffs do not correspond to any *ex interim* assessment of any agent, that the Nash equilibria are meaningful.

Other computational techniques exist for Bayesian games that also have temporal structure—that is, for Bayesian games written using the "extensive form with chance moves" formulation, for which the game tree is smaller than its induced normal form. First, there is an algorithm for Bayesian games of perfect information that generalizes backward induction (defined in Section 5.1.4), is called *expectimax*. Intuitively, this algorithm is very much like the standard backward induction algorithm given in Figure 5.6. Like that algorithm, expectimax recursively explores a game tree, labeling each non-leaf node $h$ with a payoff vector by examining the labels of each of $h$'s child nodes—the actual payoffs when these child nodes are leaf nodes—and keeping the payoff vector in which the agent who moves at $h$ achieves maximal utility. The new wrinkle is that chance nodes must also receive labels. Expectimax labels a chance node $h$ with a weighted sum of the labels of its child nodes, where the weights are the probabilities that each child node will be selected. The same idea of labeling chance nodes with the expected value of the next node's label can also be applied to extend the minimax algorithm (from which expectimax gets its name) and alpha-beta pruning (see Figure 5.7) in order to solve zero-sum games. This is a popular algorithmic framework for building computer players for perfect-information games of chance such as Backgammon.

expectimax algorithm

There are also efficient computational techniques for computing sample equilibria of imperfect-information extensive-form games with chance nodes. In particular, all the computational results for computing with the sequence form that we discussed in Section 5.2.3 still hold when chance nodes are added. Intuitively, the only change we need to make is to replace our definition of the payoff function (Definition 5.2.7) with an expected payoff that supplies the expected value, ranging over Nature's possible actions, of the payoff the agent would achieve by following a given sequence. This means that we can sometimes achieve a substantial computational savings by working with the extensive-form representation of a Bayesian game, rather than considering the game's induced normal form.

### 6.3.4 Ex post *equilibrium*

Finally, working with *ex post* utilities allows us to define an equilibrium concept that is stronger than the Bayes–Nash equilibrium.

*ex post* equilibrium

**Definition 6.3.8 (*Ex post* equilibrium)** *An* ex post *equilibrium is a mixed-strategy profile s that satisfies* $\forall\theta,\ \forall i,\ s_i \in \arg\max_{s_i' \in S_i} EU_i(s_i', s_{-i}, \theta)$.

Observe that this definition does not presume that each agent actually *does* know the others' types; instead, it says that no agent would ever want to deviate from his mixed strategy *even if* he knew the complete type vector $\theta$. This form of equilibrium is appealing because it is unaffected by perturbations in the type distribution $p(\theta)$. Said another way, an *ex post* equilibrium does not ever require any agent to believe that the others have accurate beliefs about his own type distribution. (Note that a standard Bayes–Nash equilibrium *can* imply this requirement.) The *ex post* equilibrium is thus similar in flavor to equilibria in dominant strategies, which do not require agents to believe that other agents act rationally.

Indeed, many dominant strategy equilibria are also *ex post* equilibria, making it easy to believe that this relationship always holds. In fact, it does not, as the following example shows. Consider a two-player Bayesian game where each agent has two actions and two corresponding types ($\forall_{i \in N},\ A_i = \Theta_i = \{H, L\}$) distributed uniformly ($\forall_{i \in N},\ P(\theta_i = H) = 0.5$), and with the same utility function for each agent $i$:

$$u_i(a, \theta) = \begin{cases} 10 & a_i = \theta_{-i} = \theta_i; \\ 2 & a_i = \theta_{-i} \neq \theta_i; \\ 0 & \text{otherwise.} \end{cases}$$

In this game, each agent has a dominant strategy of choosing the action that corresponds to his type, $a_i = \theta_i$. An equilibrium in these dominant strategies is not *ex post* because if either agent knew the other's type, he would prefer to deviate to playing the strategy that corresponds to the other agent's type, $a_i = \theta_{-i}$.

Unfortunately, another sense in which *ex post* equilibria are in fact similar to equilibria in dominant strategies is that neither kind of equilibrium is guaranteed to exist.

Finally, we note that the term "*ex post* equilibrium" has been used in several different ways in the literature. One alternate usage requires that each agent's strategy constitute a best response not only to every possible *type* of the others, but also to every *pure strategy profile* that can be realized given the others' mixed strategies. (Indeed, this solution concept has also been applied in settings where there is no uncertainty about agents' types.) A third usage even more stringently requires that no agent ever play a mixed strategy. Both of these definitions can be useful, e.g., in the context of mechanism design (see Chapter 10). However, the advantage of Definition 6.3.8 is that of the three, it describes the most general prior-free equilibrium concept for Bayesian games.

## 6.4 Congestion games

Congestion games are a restricted class of games that are useful for modeling some important real-world settings and that also have attractive theoretical properties. Intuitively, they simplify the representation of a game by imposing constraints on the effects that a single agent's action can have on other agents' utilities.

### 6.4.1 *Definition*

Intuitively, in a congestion game each player chooses some subset from a set of resources, and the cost of each resource depends on the number of other agents who select it. Formally, a congestion game is single-shot $n$-player game, defined as follows.

congestion game

**Definition 6.4.1 (Congestion game)** *A* congestion game *is a tuple* $(N, R, A, c)$*, where*

- $N$ *is a set of* $n$ *agents;*
- $R$ *is a set of* $r$ resources*;*