So far we have focused exclusively on static interactions. We now turn to dynamic interactions. Recall that each player in a static interaction has a single decision to make and makes it in ignorance of what the other players are doing. In a dynamic interaction, at least one player gets to see what one or more players have done before deciding what to do.

A key difference between static and dynamic interactions is that actors cannot make threats or promises in static interactions, but they can in a dynamic interaction. The very nature of a threat or promise requires that the threatener or promiser be able to see what the other actor does in order to decide whether to reward or punish the other actor. This cannot be done in a static interaction. Threats and promises and whether or not they are credible therefore play no role in static interactions. They do, however, play very important roles in analyzing dynamic interactions, finding equilibria, and making predictions about what will happen.

Before we can analyze dynamic interactions we must first see how to model them as a game. That is what this lecture does.

8.1 Extensive forms.

The most natural way to model a static interaction is as a game in strategic form. The most natural way to represent a dynamic interaction is as a game in extensive form with a game tree. (We will see why these two types of interaction are most naturally represented in these different ways when we examine the relation between strategic- and extensive-form games in the next lecture.)

The extensive form of a game shows all of the different ways that the interaction could unfold and what each actor knows as play unfolds. More specifically an extensive form game specifies five things:

1. Who the actors are and the order in which they move;
2. What actions or alternatives are available to an actor when making a decision;

*These are lecture notes for PS135/Econ110 at UC Berkeley by Robert Powell.
3. What each actor knows when making a decision;
4. A ranking or preference ordering over all of the possible ways that the game could unfold;
5. The probabilities of all *exogenous* events, i.e., events that are beyond the players’ control.

Finally, these five things are assumed to be common knowledge: Every player knows them, every player knows that every other player knows them, and so on. We will define a tree more formally below after working through several examples which help to make the five elements of a tree clearer and more concrete. Each example below begins with a description of the interaction which will then be formalized as a game in extensive form.

### 8.2 Examples of Perfect-Information Games.

A game has **perfect information** if each actor when making a decision is sure of exactly what has already happened, i.e., what actions have already been taken, by whom, and in what order. The first five examples are all perfect-information games.

*Example 1: The basic deterrence game.* There are two players, a challenger, $C$, and a defender, $D$. The challenger begins by deciding whether to accept the status quo or challenge it. The game ends if there is no challenge and the status quo remains in place. If there is a challenge, then the defender has to decide what to do. It can acquiesce to the challenge or fight. Either way, the game ends after the defender’s decision. The interaction is dynamic because the defender knows what the challenger did when the defender is deciding what to do.

The game tree for this interaction is shown in Figure 8.1. It begins with a decision node at which $C$ chooses whether to *challenge* the status quo or *accept* it. Each alternative is represented by a different branch. The game ends if the $C$ chooses branch *accept*. If $C$ challenges the status quo, there is another decision node at which $D$ chooses to *fight* or *acquiesce*.

There are three different paths through the tree, each ending in a different terminal
8.2 EXAMPLES OF PERFECT-INFORMATION GAMES.

Figure 8.1: The basic deterrence game.

**node.** Each terminal node corresponds to a different way that play could unfold. They are: (i) accept, (ii) challenge followed by acquiesce, and (iii) challenge followed by fight.

Actors have preferences over how the game ends, i.e., over the terminal nodes. These payoffs are shown in Figure 9.1 where the first number is C’s payoff and the second number is D’s. The best outcome from C’s perspective is a challenge followed by acquiesces. Next best is to accept the status quo. Worst is to for a challenge to result in a fight. The best outcome from the defender’s perspective is not to be challenged. The worst is for the interaction to end in a fight. The intermediate outcome for the defender is for there to be a challenge followed by acquiescence.

The payoffs in Figure 8.1 are consistent with this ranking and are based on a market-entry version of the deterrence game. The actors in this version are two firms. The defender is an incumbent that currently has the entire a market to itself. The challenger is a potential entrant that is considering challenging the incumbent by entering the market in the hope of splitting the profits. The total profits from the market are 2. The incumbent gets all of the profits if the potential entrant decides not to enter, i.e., accept the status quo. If the potential entrant does enter, the two firms divide the profits with each getting 1 if the incumbent acquiesces. If the incumbent fights by starting a price war, both firms lose money and get a payoff of -1.

The basic deterrence game can also be seen as a simple model of international conflict over the existing territorial status quo, say over Ukraine. The actors are states, and the challenger can accept the territorial status quo or challenge. If the defender acquiesces,

---

1A common labeling convention is that the payoff listed first belongs to the player that moves first in the tree.
Example 2: An ultimatum game. Players I and II are dividing $3. Player I can make an offer of 1, 2, or 3 dollars to player II who can accept or reject the offer. If II accepts, II gets what was offered and I pockets the remainder. If II rejects, both get zero. Each player’s payoff is the amount of money it has at the end.

The game tree is shown in Figure 8.2. The tree begins with a decision node for I with a branch for each possible offer. At the end of each branch, II chooses whether to accept or reject the offer. Player I can make 3 different offers and II can respond in two different ways to each offer. So there are $3 \times 2 = 6$ possible ways that events could unfold, i.e., six different paths through the tree.

Example 3: A pay-raise game with public voting. A committee composed of three members, A, B, and C, has to decide whether to give itself a pay raise. Each member can vote yes or no, and they get a raise if a majority, i.e., two or more members, votes in
favors of the raise. Voting is public. The committee secretary calls on A first. After she
has announced her vote, the secretary calls on B followed by C.

As for the members’ payoffs, voting oneself a raise can be problematic. On the one
hand, one may want or need a raise. On the other hand, one may not want to be seen
to be voting oneself a raise, especially if one is a member of Congress or a state legislator
who may be concerned about what his constituents may think. With this in mind, there
are four possible outcomes. The best outcome, with a payoff of 4, is to get the raise but
to have voted against it. Next best with a payoff of 3 is to get the raise after having voted
for it. After that with a payoff of 2 is to not get the raise but at least to have also voted
against it. Worst of all, is to not get a raise after having voted for it.

The tree is illustrated in Figure 8.3. Consider the red path through the tree. This
8.2 EXAMPLES OF PERFECT-INFORMATION GAMES.

![Tic-tac-toe grid](image)

Figure 8.4: Tic-tac-toe grid.

represents one way that events could unfold, namely, A votes yes, B votes no, and C votes yes. If history unfolds this way, the raise passes because a majority composed of A and C vote for it. Each of them gets a payoff of 3. B however gets both the raise and the benefits of having voted against it which yields a payoff of 4.

We will solve this game later. Suppose for now that you could choose whether to vote first, second, or third, i.e., you can choose to be player A, B, or C. Who would you want to be?

**Example 4: Tic-tac-toe or X’s and O’s.** Many people learn to play tic-tac-toe as a child. The game is played on a $3 \times 3$ grid like that in Figure 8.4 where each cell in the grid has been labeled for convenience. X begins by putting an x in any one of the nine cells. O then puts an o in one of the eight remaining open cells. The game continues in this way with the players taking turns until X wins by getting 3 x’s in a horizontal, vertical, or diagonal row or O wins by getting three o’s in a row. Take the payoffs to be 1 for winning, -1 for losing, and zero for a draw.

This game has a lot of branches though with a few hours (and a lot of patience) one could draw it. There are nine ways for X to start, and O can follow each of these in eight different ways. So there are seventy-two different ways the first round of moves can be played, and some of these are shown in Figure 8.5. The quickest way that the game can end is for X to get three in a row as fast as possible, i.e., after just three moves. Hence
8.2 Examples of Perfect-Information Games.

The shortest path through the tree is five branches long, three for $X$ and two for $O$. There are $15,120 = 9 \times 8 \times 7 \times 6 \times 5$ different ways that the first five rounds can be played (though not all of them result in $X$’s winning). Trees can get very big very quickly even for very simple games.

Example 5: Chess. Tic-tac-toe is a very simple game which most people “solve” after playing it a few times, i.e., most find the optimal way to play. (If both players play optimally, the game ends in a draw.) Chess by contrast is far from simple. Assuming that the game is declared a stalemate if no one has won after 50 moves, then it is possible in principal to draw the game tree. That is, there is a finite number of players, namely two, Black and White. Every time a player has to decide what to do, it has a finite number of alternatives. (Each player has 16 or fewer pieces and each piece can only move in a
8.3 Examples of Imperfect-Information Games.

A game has **imperfect information** if one or more actors ever has to make a decision without knowing exactly what has happened prior to that decision. One actor, for example, is unsure of what action another actor took. The pay-raise game with secret balloting instead of public voting is a game of imperfect information.

*Example 6: The pay-raise game with secret ballots.* Suppose the voting in the pay raise

\[2\text{Though finite, the number is huge. By some estimates, the number of branches in the tree is larger than the number of atoms in the observable universe.}\]
8.3 EXAMPLES OF IMPERFECT-INFORMATION GAMES.

game is done by secret ballot instead of public roll call. A still votes first but does so by writing her vote on a ballot which she seals in an envelope and passes to the committee secretary. B then passes his secret ballot to the secretary and C follows. The key thing here is that even though the balloting is happening sequentially in chronological time, no one knows how the other two have voted when casting one’s own vote.⁴

Compared to the public voting version of the game, the sequence of moves is the same; the alternatives are the same, namely, vote yes or no, and the payoffs are the same. The only thing that is different is what the members know when voting. For example, when B decides how to vote, he knows how A voted when there is public voting but not when ballots are secret. Similarly, C knows how both A and B voted with public voting but not with secret ballots. Figure 8.6 shows how this is represented in a tree.

Consider B’s decision. He does not know if he it is at the top decision node, i.e., A voted yes, or the bottom node. C when deciding how to vote does not know if she is at her top node, that is, both A and B voted yes; the bottom node because both voted no; or one of the other two nodes. The inability to distinguish among nodes is typically represented in one of two ways. The first is by enclosing the nodes in an oval as B’s are. The second is linking the indistinguishable nodes with a dashed line as C’s are.

An an information set is composed of the decision nodes that a player cannot distinguish from each other but can distinguish from others in the tree. B’s two decision nodes are in the same information set. B is at the top node if A voted ”yes” and at the bottom node if A voted “no.” But B does not know how A voted and therefore cannot distinguish between these two nodes. Similarly, all of C’s decision nodes are in a single information set. What makes one node different from another are the votes leading up to. C however does not know how A and B voted and consequently cannot distinguish any one decision node from another.

Example 7: The pay-raise game with semi-secret balloting. How do you know whether two nodes should be in the same information set or in different information sets? Suppose

⁴This of course makes the interaction static. Each actor has a single decision to make and makes it in ignorance of what other actors have decided to do. We discuss the relation between strategic-form and extensive-form representations in the next lecture.
the (bizarre) rules of the committee are that $A$ votes secretly and then $B$ votes publicly. How is this information structure represented?

As before, $B$ does not know how $A$ voted. Because $B$ cannot distinguish between its two decision nodes, they should be included in the same information set as illustrated in Figure 8.7. Now consider $C$'s nodes [1] and [2]. Node [1] corresponds to the history in which $A$ voted $yes$ and $B$ voted $yes$. The history leading up to [2] is a $yes$ from $A$ and a $no$ from $B$. What makes these two paths different is how $B$ voted. Because $C$ hears how $B$ votes, she can distinguish the path leading up to [1] from that leading to [2]. Hence, these two nodes should be in different information sets.

By contrast, [1] and [3] are in the same information set. What differentiates the paths leading to these nodes is how $A$ votes. But $A$ casts her ballot in secret, so $C$ does not know how she voted and cannot distinguish between the path leading to [1] from the
path leading to [3]. Continuing in this way shows that $C$ has two information sets, one composed of decision nodes [1] and [3] and the other comprised of decision nodes [2] and [4].

In sum, two decision nodes $[a]$ and $[b]$ at which player $P$ is making a decision are in different information sets if two conditions hold. First, if $P$ has different alternatives at each node, then $P$ can distinguish between these nodes on that basis and the nodes are not in the same information set. Second, if $P$ can observe anything that makes the path leading to $[a]$ different from the path leading to $[b]$, then $P$ can distinguish the two nodes and they are not in the same information set.
8.4 Perfect or Imperfect Information?

When looking at a tree, it is very easy to see if it is a game of perfect or imperfect information. If every information set has one and only one decision node in it, then the game has perfect information. Whenever a player makes a decision, it is certain of the history (path through the tree) leading up to that decision. If one or more information sets has two or more decision nodes, the game has imperfect information. When a player’s information set has two or more nodes, that player is unsure of precisely which path prior play has taken.

8.5 Events Beyond the Actors’ Control.

*Example 8: Very simple poker.* A final example includes an exogenous event, i.e., an event that is beyond the actors’ control. The game is a very, very simple version of poker. There are two players, I and II, and each has already put a dollar in the pot. The game begins with the deal of the cards. One card is dealt to each player. Each players sees his or her card but not the other player’s card.

After the cards are dealt, I decides where to bid or fold. If I folds, the game ends and II gets the pot. This gives I a payoff of -1, namely the dollar he put in the pot, and II a payoff of 1, the pot of two less the dollar she originally put in the pot. If I bids, he adds a second dollar to the pot and it is II’s turn to decide what to do.

II can either fold or call. If she folds, the game ends and the pot is divided with I getting a net payoff of 1 and II getting a payoff of -1. If II calls, she puts a second dollar into the pot and both players turn over their cards.

The winner is determined by the color of the card. A black card beats a red card with the player holding the black card getting the pot and hence a net payoff of 2. The loser’s payoff is -2. If the cards are the same color, the game is a tie. Each player gets his or her money back and has a net payoff of 0.

The first thing that happens in the game is that the cards are dealt. There are four possible ways of dealing one card to each player: a black to both denoted by (B, B) where the first letter denote’s the color of I’s card; a black to I and a red to II, (B, R); a red to
both, \((R, R)\); and a red to \(I\) and black to \(II\), \((R, B)\).

Which of these combinations is actually dealt is an exogenous event beyond the actors’ control. To represent events like this in a tree, we use a modeling device. We assume that there is a fictitious player called “Nature” or \(N\) that actually chooses the hand. But Nature does not have preferences over outcomes. Rather, Nature does things according to probabilities determined by the situation we are trying to model. These probabilities are the fifth thing that a tree must specify.

The probability of dealing two black cards, \((B, B)\), is the probability that the first card is black times the probability that the second card is black. Since there are 26 black cards in a normal deck of 52 cards, the probability that the first card dealt is \(26/52 = 1/2\). Since there are now 51 cards left in the deck and 25 of them are black, the chances that the second card is black is \(25/51\). This means that probability of \((B, B)\) is \((1/2)(25/51) = 25/102\) which is slightly less than a fourth. The probabilities of the other combinations are derived in the same way and shown in Figure 8.8.

Figure 8.8 shows the sequence of moves. \(N\) deals the hand. \(I\) then decides whether to bid or fold. If \(I\) bids, \(II\) chooses whether to fold or call. The figure also shows how the players rank the terminal nodes.
Figure 8.9: The simple-poker game.

Figure 8.8 does not show the correct information structure. The game in the figure has perfect information. I is holding a black card at nodes [1] and [2]. But these nodes are in different information sets which means that this tree represents an interaction in which I can distinguish the deal (B, B) from (B, R). That is, I knows the color of the card dealt to II. This might be the correct information structure if the cards were marked. But that is not the situation we are trying to model. Rather, I cannot distinguish between (B, B) and (B, R), so nodes [1] and [2] should be in the same information set. The correct information structure is shown in Figure 8.9 and you should be able to verify that it is the correct structure.

8.6 A More Technical Description of an Extensive Form.

The description of an extensive form given above in section 8.1 is sufficient for much applied work and the rest of this book. This chapter concludes with a more technical formulation for those who plan on doing future work in game theory and need or want to see a more precise definition. A tree formally consists of:

1. A set of $K$ players plus Nature or $N$. 
2. A set of nodes $Z$ with a designated root $z_0 \in Z$.

3. A precedence relation $\succ$ defined over the nodes. If $z \succ z'$, then $z$ is said to precede $z'$ or, equivalently, that $z$ is on the path to $z'$. This relation describes which nodes follow which other nodes along the various paths through the tree. The relation satisfies four properties:
   
   (a) The root $z_0$ precedes every other node: $z_0 \succ z$ for all $z \neq z_0$ and $z \in Z$.

   (b) The relation is transitive: If $x$ precedes $y$ ($x \succ y$) and $y$ is on the path to $z$ ($y \succ z$), then $x$ is on the path to $z$ ($x \succ z$).

   (c) The relation is anti-symmetric: If $x$ precedes $y$, then $y$ does not precede $x$: $x \succ y$ implies $y \not\succ x$.

   (d) The relation is partial: There are nodes $x$ and $y$ such that $x$ is not on the path to $y$ ($x \not\succ y$) and $y$ is not on the path to $x$ ($y \not\succ x$). This means that there are at least two paths through the tree.

4. *Terminal nodes* are nodes in $Z$ that do not precede any other nodes: $t$ is a terminal node if $t \not\succ z$ for any $z \in Z$. Nodes that are not terminal nodes are *decision nodes*.

Parts (2)-(4) describe the tree. Parts (5) and (6) label those parts. Part (5) associates each decision node with a player, and part (6) names the branches at each decision node.

6. There is a “player partition” of the decision nodes which associates each decision node with one and only one of the $K + 1$ players.

7. To label the branches, let $A$ be the set of names of all of the possible actions that can be taken in the game. For the deterrence game, $A = \{\text{accept, challenge, acquiesence, fight}\}$. For the pay-raise game, $A = \{\text{yes, no}\}$. As for the branches themselves, a branch is defined by the two nodes it connects. That is, there is a branch from $x$ to $y$ if $x$ precedes $y$ and no other node is between them, i.e., there is no $z$ such that $x \succ z \succ y$. Let $B$ be the set of branches and $B(x)$ be the set of branches emanating from $x$. The branches get names by assuming there is a function that assigns every branch to an element of $A$. Let $A(x)$ be the names assigned to the branches emanating from $x$. 
Now that the nodes have been assigned to the players and the alternatives at each node have been named, the information structure can be specified.

8. There is an “information partition” of each player’s decision nodes specifying which nodes belong to the same information set and which nodes belong to different information sets. Every node belongs to an information set. Let $I(x)$ be the information set containing $x$. If no other nodes belong to $x$ then $I(x)$ is a singleton. More substantively, the actor deciding what to do at $x$ knows exactly how play has unfolded in the run up to its decision at $x$. If two nodes $x$ and $y$ are in the same information set, then $I(x) = I(y)$. Another required condition is that the player must have the same alternatives at those nodes: $A(x) = A(y)$.

9. The probabilities of exogenous events must be given. If $n$ is a decision node belonging to Nature, then there is a probability distribution over the actions that can be taken at that node, namely, $A(n)$. Since Nature must take some action at $n$, the probabilities at $A(n)$ must sum to one.

Parts (1)-(9) define the physical environment. The last part formalizes the players’ motivations.

10. Players have utility functions defined over the terminal nodes of the tree.

Having a formal description is essential for proving some general results but those results are beyond the scope of this course.