The Games We Play

Jacob W. Crandall

September 10, 2012

The various disciplines studying multi-agent choice, including game theory and its variants, multi-agent systems, AI, etc., rely on our ability to model real-world situations with games. The purpose of these notes are to cover some of the basics of the games we will/can/should consider as we study multi-agent learning. Specifically, we will talk about the different kinds of games used in the literature and the characteristics of these games.

1 Kinds of Games

We briefly cover normal-form games, extensive-form games, and stochastic games.

1.1 Normal-Form Games

Normal-form games are also known as canonical games or matrix games. We will mostly just use the term matrix game throughout the class. A complete matrix game can be displayed (at least for 2-player[1] games) as a single payoff matrix. Each cell of the payoff matrix specifies the payoffs to each player given the actions of the all the players in the game. For a 2-player game, we assign the rows to the actions of one of the players (called ROW) and the columns to the other player (called COL).

For example, consider the following payoff matrix:

	`Left`	`Right`
`Up`	1, 3	-1, 2
`Down`	2, -1	2, 0

In this matrix, ROW can play either `Up` or `Down`. Likewise, COL has two possible actions: `Left` or `Right`. In each cell of the payoff matrix, the payoffs to ROW are listed first, followed by the payoffs to COL. Thus, if ROW plays `Up` and COL plays `Right`, then ROW gets a payoff of -1, and COL gets a payoff of 2.

A matrix game can be either a simultaneous or a turn-taking game. In a *simultaneous-move game*, each player chooses its action without knowledge of the other player's action (both actions are executed simultaneously). In a *turn-taking game*, one of the players must select its action first. The other player (or players, as the case may be) observe this action and then decide and execute its action.

1.2 Extensive-Form Games

A special thanks to Professor Michael Goodrich at Brigham Young University, who graciously allowed me to lift this content directly from his web site. I have modified the content somewhat, though some of it is taken directly in his words. The figure also comes from his web site.
`http://students.cs.byu.edu/~cs670ta/ExtensiveForm.html`

We can model many interesting problems with a matrix game. However, you are probably thinking that many situations require an outcome to be based on sequences of moves (or actions) by the agents involved

[1] Throughout these notes (and throughout the class), we will use the terms *agent* and *player* interchangeably.

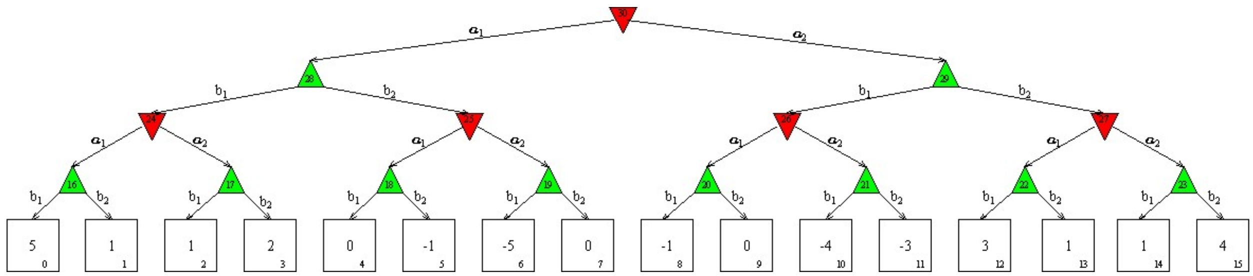

Figure 1: An example of an extensive-form game.

rather than just a single move. *Extensive-form games* are able to capture such situations. An example of an extensive form game is shown in Figure 1.

This game, which is a turn-taking game, starts with the red agent (denoted in the tree by a red triangle pointing down) choosing either action a_i or a_2 (for lack of a better name we will just call this player DOWN-A, since it choices the A action and it is represented with a triangle pointing down). The green agent (denoted in the tree by a green triangle pointing up) then chooses either b_1 or b_2 (we call this agent UP-B). DOWN-A then selects another action, after which UP-B selects a second action. The numbers in the boxes at the leaves of the tree represent the DOWN-A's payoff for the sequence of choices. For example, if the agents choose a_1, b_1, a_1, b_1, then DOWN-A gets five dollars. Note that since this is a zero-sum game (i.e., $u_1 = -u_2$, where u_i is the payoff to player i), we need only specify the payoff to the first player (DOWN-A). Thus, in this example, the payoff to UP-B is negative five dollars.

The tree structure of this game makes it clear that the outcome depends not on a single action chosen by each of the agents, but instead on a sequence of such choices. Thus, the terminology *extensive form* represents the idea that the game extends across time or turns (tree structure). When outcomes depend only on a single choice by the agents, then the game is said to be in *normal form*, meaning the game is in a normative or canonical form (payoff matrix).

It would be nice if there were only one game form, since this would allow us to develop only one theory. We'll talk later in class about how all extensive-form games can be converted to normal-form games

1.3 Stochastic Games

An alternate world modeling tool is the *stochastic game* or Markov game, which is an extension of Markov decision processes to environments with multiple adapting agents. A stochastic game has the ability to model probabilistic transitions, and is, like extensive form games, played in a sequence of stages.

Formally, a stochastic game is a tuple (n, Σ, A, T, R), where n is the number of agents in the game, Σ is a set of states, $A = (A_1, \ldots, A_n)$ is the set of joint actions (where A_i is the set of actions available to agent i), T is a transition function $\Sigma \times A \times \Sigma \to [0, 1]$, and $R = (R_1, \ldots, R_n)$ is the reward function $\Sigma \times A \to R$ (where R_i is the reward function for agent i).

2 Game Characteristics

Regardless of the form the game takes on, it can be categorized along many dimensions. We briefly discuss three different dimensions in this section. These dimensions consist of the the competitive nature of the game, the knowledge/information afforded to the agents, and the number of times the game is repeated.

2.1 Competition vs. Cooperation

In general, game theory distinguishes between cooperative situations and non-cooperative situations. I find it useful to further break down non-cooperative situations since the nature of these two situations often require different approaches. Thus, we will discuss coordination games, constant-sum games, and games of conflicting interests. Collectively, these games are referred to as *general-sum games*.

2.1.1 Coordination Games

In coordination games, what is good for one agent is good for all agents. An example coordination game in normal form is:

	a	b
A	2, 2	0, 0
B	0, 0	1, 1

In this game, agents try to coordinate their actions. The joint action (A, a) is the most desirable, but the joint action (B, b) also produces positive payoffs to all agents. This particular game is called a *pure coordination* game since the agents always receive the same payoff.

Other coordination games move more toward the domain of games of conflicting interest (which we will discuss shortly). For example, consider the stag hunt game:

	stag	hare
stag	3, 3	0, 1
hare	1, 0	1, 1

In this game, each player can choose to hunt stag or hare. In order to catch a stag (the biggest animal, hence the bigger payoff of 3), both players must choose to hunt the stag. However, a hunter does not need help to catch a hare, which yields a payoff of 1. Thus, in general, it is best for the hunters to coordinate their efforts to hunt stag, but there is considerable risk in doing so (if the other agent decides to hunt hare). In this game, the payoffs are the same to the agents when the coordinate their actions, but their payoffs are not equal when they do no coordinate their actions.

2.1.2 Constant-Sum Games

Constant-sum games are games in which the sum of the players' payoffs for each out all sum to the same number. These games are games of pure competition in which "my gain is your loss." Zero-sum games are a specific example of these games. One such game is the game rock, paper, scissors[2], which we can conveniently put in matrix form as follows:

	rock	paper	scissors
rock	0, 0	-1, 1	1, -1
paper	1, -1	0, 0	-1, 1
scissors	-1, 1	1, -1	0, 0

2.1.3 Games of Conflicting Interests

We will refer to the wide-range of games between constant-sum games and coordination games as games of conflicting interest. In these games, the agents have somewhat opposing interests, but all players can benefit from making certain compromises. People (and learning algorithms) are often tempted to play competitively in these games (both in the real world and in games), though they often hurt themselves by doings so. However, on the other hand, taking an overly cooperative stance can lead to similarly bad (or worse) payoffs. One of the most celebrated games of this type is the prisoners' dilemma:
We will study a wide variety of games of conflicting interest throughout this class.

[2]It is interesting to note that there are actually world-wide rock, paper, scissor competitions (for both human players and computer algorithms). Cheap thrills!

	cooperate	defect
cooperate	3, 3	0, 5
defect	5, 0	1, 1

2.2 Repetition

Any of the previously mentioned kinds of games can be played any number of times between the same players. The number of rounds of the game can change the nature of the game. Throughout this class, we will refer to *one-shot games* and *repeated games*.

2.2.1 One-shot Games

In one-shot games, agents interact for only a single round (or stage). Thus, in these situations there is no possible way for agents to reciprocate (by inflicting punishment or rewards) thereafter. As a result, these games are especially useful in testing the self-interest of individuals. We will discuss this topic in detail when we discuss behavioral game theory. However, in this class we will focus more fully on situations in which the agents interact repeatedly.

2.2.2 Repeated Games

In repeated games, agents interact with each other for multiple rounds (playing the same game). In such situations, agents have opportunities to adapt to each others' behaviors (i.e., learn) in order to try to become more successful. The number of times the game is repeated can be important. For example, in *finite-horizon* repeated games the same game is repeated a fixed number of times by the same agents. After the fixed number of rounds, the interaction is terminated. *Infinite-horizon* games are games in which play is repeated indefinitely or are repeated with some probably γ after each round.

2.2.3 Dynamic Games

In most circumstances throughout this class, we will focus on situations in which the same game is repeated (usually in the infinite horizon situation). This restriction is made for simplicity. However, in real-world, the game typically changes when agents interact repeated (I will call these repeated games *dynamic games*). Hopefully, the principles and theory we study and develop in this class will help us move toward being able to develop learning algorithms for dynamic games.

2.3 Knowledge/Information

All of the kinds of games we have discussed so far can further differ by the kinds of information afforded to the agents in the game. For example, does an agent know the payoffs (or preference orderings) of other agents? Does the agent know its own payoff matrix? Can (s)he/it view the actions and payoffs of other agents? All of these (and other related) questions are important as they can help determine how the agent should learn and act. The more information an agent has about the game, the better (s)he should be able to do (theoretically). Knowing what information to use and how to use it are central issues of multi-agent/distributed learning.

	a	b
a	1	3
b	0	2

(a)

	a	b
a	1, -1	3, -3
b	0, 0	2, -2

(b)

	a	b
a	1, 1	3, 3
b	0, 0	2, 2

(c)

	a	b
a	1, 1	3, 0
b	0, 3	2, 2

(d)

Figure 2: Example showing how knowledge of one's own payoff matrix R_i does not indicate the kind of game an agent is playing. (a) Shows player 1's payoff matrix R_1, and (b)–(d) show three different kinds of games that the agents could be playing given that payoff matrix.

In short, the information an agent has about the game can vary along the following dimensions:

1. Knowledge of the agent's own actions
2. Knowledge of the agent's own payoffs
3. Knowledge of the existence of other agents
4. Knowledge of the other agents' actions
5. Knowledge of the other agents' payoffs
6. Knowledge of the other agents' internal workings (the learning algorithms they use, etc.)

Note that each level of information can be known partially or by degrees.

To illustrate the importance of information, consider Figure 2. Figure 2a shows the payoff matrix of the row agent. To know or construct this payoff matrix, it means that the agent knows its own actions as well as the actions of its associate, and its own payoffs. This information is very useful to the agent, however, without knowledge of its associate's payoffs, the agent does not know what kind of game it is playing. The game could be a fully competitive (constant-sum) game (Figure 2b), a coordination game (Figure 2c), or a game of conflicting interest (Figure 2d). Since an agent might want to play differently depending on the kind of game it is playing, these knowledge restrictions can be quite difficult to overcome. Nevertheless, a successful learning algorithm should be able to learn successfully even with incomplete knowledge (though many do not).

It is useful to know some of the terms used by game theorists when they talk about information availability. Game theorists use the term *complete information* when each player has knowledge of the payoffs and possible strategies of other players. Thus, *incomplete information* refers to situations in which the payoffs and strategies of other agents are not completely known. The term *perfect information* refers to situations in which the actual actions taken by associates are fully observable. Thus, *imperfect information* implies that the exact actions taken by associates are not fully known.

You might also hear the term *information set* thrown around. An information set refers to the amount of information available to an agent when it makes a decision at a specific point (or node) in an extensive-form game.