Reading: excerpt from "Multiagent Systems", chapter 3.
Lecture Overview

Recap

Game Theory

Example Matrix Games
Rewards and Values

Suppose the agent receives the sequence of rewards $r_1, r_2, r_3, r_4, \ldots$. What value should be assigned?

- total reward $V = \sum_{i=1}^{\infty} r_i$
- average reward $V = \lim_{n \to \infty} \frac{r_1 + \cdots + r_n}{n}$
- discounted reward $V = \sum_{i=1}^{\infty} \gamma^{i-1} r_i$
  - $\gamma$ is the discount factor
  - $0 \leq \gamma \leq 1$
Policies

- A **stationary policy** is a function:
  \[ \pi : S \rightarrow A \]

  Given a state \( s \), \( \pi(s) \) specifies what action the agent who is following \( \pi \) will do.

- An **optimal policy** is one with maximum expected value
  - we’ll focus on the case where value is defined as discounted reward.

- For an MDP with stationary dynamics and rewards with infinite or indefinite horizon, there is always an optimal stationary policy in this case.
Value of a Policy

- $Q^{\pi}(s, a)$, where $a$ is an action and $s$ is a state, is the expected value of doing $a$ in state $s$, then following policy $\pi$.
- $V^{\pi}(s)$, where $s$ is a state, is the expected value of following policy $\pi$ in state $s$.
- $Q^{\pi}$ and $V^{\pi}$ can be defined mutually recursively:

$$V^{\pi}(s) = Q^{\pi}(s, \pi(s))$$
$$Q^{\pi}(s, a) = \sum_{s'} P(s'|a, s) \left( r(s, a, s') + \gamma V^{\pi}(s') \right)$$
Value of the Optimal Policy

- $Q^*(s, a)$, where $a$ is an action and $s$ is a state, is the expected value of doing $a$ in state $s$, then following the optimal policy.
- $V^*(s)$, where $s$ is a state, is the expected value of following the optimal policy in state $s$.
- $Q^*$ and $V^*$ can be defined mutually recursively:

\[
Q^*(s, a) = \sum_{s'} P(s'|a, s) \left( r(s, a, s') + \gamma V^*(s') \right)
\]

\[
V^*(s) = \max_a Q^*(s, a)
\]

\[
\pi^*(s) = \arg \max_a Q^*(s, a)
\]
**Value Iteration**

- **Idea:** Given an estimate of the $k$-step lookahead value function, determine the $k + 1$ step lookahead value function.
- **Set $V_0$ arbitrarily.**
  - e.g., zeros
- **Compute $Q_{i+1}$ and $V_{i+1}$ from $V_i$:**
  
  $$Q_{i+1}(s, a) = \sum_{s'} P(s'|a, s) \left( r(s, a, s') + \gamma V_i(s') \right)$$
  $$V_{i+1}(s) = \max_a Q_{i+1}(s, a)$$

- **If we intersect these equations at $Q_{i+1}$, we get an update equation for $V$:**
  
  $$V_{i+1}(s) = \max_a \sum_{s'} P(s'|a, s) \left( r(s, a, s') + \gamma V_i(s') \right)$$
Asynchronous VI: storing $Q[s, a]$

- **Repeat forever:**
  - Select state $s$, action $a$;
  - $Q[s, a] \leftarrow \sum_{s'} P(s'|s, a) \left( R(s, a, s') + \gamma \max_{a'} Q[s', a'] \right)$;
Lecture Overview

Recap

Game Theory

Example Matrix Games
Non-Cooperative Game Theory

- What is it?
Non-Cooperative Game Theory

What is it?
- mathematical study of interaction between rational, self-interested agents
Non-Cooperative Game Theory

What is it?
- mathematical study of interaction between rational, self-interested agents

Why is it called non-cooperative?
Non-Cooperative Game Theory

- What is it?
  - mathematical study of interaction between rational, self-interested agents

- Why is it called non-cooperative?
  - while it’s most interested in situations where agents’ interests conflict, it’s not restricted to these settings
  - the key is that the individual is the basic modeling unit, and that individuals pursue their own interests
    - cooperative/coalitional game theory has teams as the central unit, rather than agents

- You can think of a non-cooperative game as a decision diagram where different agents control different decision nodes, and where each agent has his own utility node.
Should you send your packets using correctly-implemented TCP (which has a “backoff” mechanism) or using a defective implementation (which doesn’t)?

- Consider this situation as a two-player game:
  - both use a correct implementation: both get 1 ms delay
  - one correct, one defective: 4 ms delay for correct, 0 ms for defective
  - both defective: both get a 3 ms delay.
TCP Backoff Game

- Consider this situation as a two-player game:
  - both use a correct implementation: both get 1 ms delay
  - one correct, one defective: 4 ms delay for correct, 0 ms for defective
  - both defective: both get a 3 ms delay.

- Questions:
  - What action should a player of the game take?
  - Would all users behave the same in this scenario?
  - What global patterns of behaviour should the system designer expect?
  - Under what changes to the delay numbers would behavior be the same?
  - What effect would communication have?
  - Repetitions? (finite? infinite?)
  - Does it matter if I believe that my opponent is rational?
Defining Games

- Finite, \( n \)-person game: \( \langle N, A, u \rangle \):
  - \( N \) is a finite set of \( n \) players, indexed by \( i \)
  - \( A = A_1, \ldots, A_n \) is a set of actions for each player \( i \)
    - \( a \in A \) is an action profile
  - \( u = \{u_1, \ldots, u_n\} \), a utility function for each player, where \( u_i : A \mapsto \mathbb{R} \)

- Writing a 2-player game as a matrix:
  - row player is player 1, column player is player 2
  - rows are actions \( a \in A_1 \), columns are \( a' \in A_2 \)
  - cells are outcomes, written as a tuple of utility values for each player
Lecture Overview

Recap

Game Theory

Example Matrix Games
Here’s the **TCP Backoff Game** written as a matrix ("normal form") and as a decision network.

<table>
<thead>
<tr>
<th></th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>C</strong></td>
<td>−1,−1</td>
<td>−4,0</td>
</tr>
<tr>
<td><strong>D</strong></td>
<td>0,−4</td>
<td>−3,−3</td>
</tr>
</tbody>
</table>

![Action by Player 1](P1_Utility) ![Action by Player 2](P2_Utility)
Games in Matrix Form

Here’s the **TCP Backoff Game** written as a matrix (“normal form”) and as a decision network.

<table>
<thead>
<tr>
<th></th>
<th><strong>C</strong></th>
<th><strong>D</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>C</strong></td>
<td>−1, −1</td>
<td>−4, 0</td>
</tr>
<tr>
<td><strong>D</strong></td>
<td>0, −4</td>
<td>−3, −3</td>
</tr>
</tbody>
</table>

Play this game with someone near you, repeating five times.
Prisoner’s dilemma is any game

\[
\begin{array}{cc}
C & D \\
C & a, a & b, c \\
D & c, b & d, d \\
\end{array}
\]

with \( c > a > d > b \).
Players have **exactly opposed** interests

- There must be precisely two players (otherwise they can’t have exactly opposed interests)
- For all action profiles \( a \in A \), \( u_1(a) + u_2(a) = c \) for some constant \( c \)
  - Special case: zero sum
- Thus, we only need to store a utility function for one player
### Matching Pennies

One player wants to **match**; the other wants to **mismatch**.

<table>
<thead>
<tr>
<th></th>
<th>Heads</th>
<th>Tails</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heads</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>Tails</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>

The popular children's game of **Rock, Paper, Scissors**, also known as **Rochambeau**, Rock, Paper, Scissors, or Rochambeau game provides a three-strategy generalization of the matching-pennies game. The payoff matrix of this zero-sum game is shown in Figure 3.6. 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, each of the actions wins over one of the other actions, and loses to the other remaining action.
One player wants to *match*; the other wants to *mismatch*.

<table>
<thead>
<tr>
<th></th>
<th>Heads</th>
<th>Tails</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heads</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>Tails</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>

Play this game with someone near you, repeating five times.
Rock-Paper-Scissors

Generalized matching pennies.

<table>
<thead>
<tr>
<th></th>
<th>Rock</th>
<th>Paper</th>
<th>Scissors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rock</td>
<td>0</td>
<td>-1</td>
<td>1</td>
</tr>
<tr>
<td>Paper</td>
<td>1</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>Scissors</td>
<td>-1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

...Believe it or not, there's an annual international competition for this game!
Players have exactly the same interests.

- no conflict: all players want the same things
- ∀a ∈ A, ∀i, j, ui(a) = uj(a)
- we often write such games with a single payoff per cell
- why are such games “noncooperative”? 
Coordination Game

Which side of the road should you drive on?

<table>
<thead>
<tr>
<th></th>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Right</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
Recap

Game Theory

Example Matrix Games

Coordination Game

Which side of the road should you drive on?

<table>
<thead>
<tr>
<th></th>
<th>Left</th>
<th>Right</th>
</tr>
</thead>
<tbody>
<tr>
<td>Left</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Right</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Play this game with someone near you, repeating five times.
General Games: Battle of the Sexes

The most interesting games combine elements of cooperation and competition.

<table>
<thead>
<tr>
<th></th>
<th>B</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>B</td>
<td>2,1</td>
<td>0,0</td>
</tr>
<tr>
<td>F</td>
<td>0,0</td>
<td>1,2</td>
</tr>
</tbody>
</table>
General Games: Battle of the Sexes

The most interesting games combine elements of cooperation and competition.

\[
\begin{array}{cc}
B & F \\
B & 2,1 & 0,0 \\
F & 0,0 & 1,2
\end{array}
\]

Play this game with someone near you, repeating five times.