CSE 473: Artificial Intelligence
Expectimax, Uncertainty, Utilities

Dieter Fox

[These slides were created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley.
All CS188 materials are available at http://ai.berkeley.edu.]

Worst-Case vs. Average Case

Idea: Uncertain outcomes controlled by chance, not an adversary!

Expectimax Search

- Why wouldn’t we know what the result of an action will be?
  - Explicit randomness: rolling dice
  - Unpredictable opponents: the ghosts respond randomly
  - Actions can fail: when moving a robot, wheels might slip
- Values should now reflect average-case (expectimax) outcomes, not worst-case (minimax) outcomes
- Expectimax search: compute the average score under optimal play
  - Max nodes as in minimax search
  - Chance nodes are like min nodes but the outcome is uncertain
  - Calculate their expected utilities
  - i.e. take weighted average (expectation) of children
- Later, we’ll learn how to formalize the underlying uncertain-result problems as Markov Decision Processes

Minimax vs Expectimax

Expectimax

3 ply look ahead, ghosts move randomly

Minimax

Expectimax Pseudocode

```python
def value(state):
    if the state is a terminal state: return the state’s utility
    if the next agent is MAX: return max-value(state)
    if the next agent is EXP: return exp-value(state)

def max-value(state):
    initialize v = -∞
    for each successor of state:
        v = max(v, value(successor))
    return v

def exp-value(state):
    initialize v = 0
    for each successor of state:
        p = probability(successor)
        v = p * value(successor)
    return v
```
### Expectimax Pseudocode

```
def exp_value(state):
    initialize v = 0
    for each successor of state:
        p = probability(successor)
        v += p * value(successor)
    return v
```

### Expectimax Example

\[ v = \frac{1}{2} (8) + \frac{1}{3} (24) + \frac{1}{6} (-12) = 10 \]

### Expectimax Pruning?

### Depth-Limited Expectimax

### Probabilities

- A random variable represents an event whose outcome is unknown.
- A probability distribution is an assignment of weights to outcomes.

Example: Traffic on freeway.
- Random variable: \( T \) = whether there’s traffic.
- Outcomes: \( T \) in \{none, light, heavy\}.
- Distribution: \( P(T=none) = 0.25 \), \( P(T=light) = 0.50 \), \( P(T=heavy) = 0.25 \).

Some laws of probability (more later):
- Probabilities are always non-negative.
- Probabilities over all possible outcomes sum to one.

As we get more evidence, probabilities may change:
- \( P(T=heavy) = 0.25 \).
- \( P(T=heavy | \text{witness}) = 0.40 \).

We’ll talk about methods for reasoning and updating probabilities later.

### Reminder: Probabilities

- 0.25
- 0.50
- 0.25
Reminder: Expectations

- The expected value of a function of a random variable is the average, weighted by the probability distribution over outcomes.
- Example: How long to get to the airport?

<table>
<thead>
<tr>
<th>Time</th>
<th>Probability</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>20 min</td>
<td>0.25</td>
<td>+</td>
</tr>
<tr>
<td>30 min</td>
<td>0.50</td>
<td>+</td>
</tr>
<tr>
<td>60 min</td>
<td>0.25</td>
<td>+</td>
</tr>
<tr>
<td></td>
<td></td>
<td>= 35 min</td>
</tr>
</tbody>
</table>

What Probabilities to Use?

- In expectimax search, we have a probabilistic model of how the opponent (or environment) will behave in any state.
  - Model could be a simple uniform distribution (roll a die).
  - Model could be sophisticated and require a great deal of computation.
  - We have a chance node for any outcome out of our control or the opponent or environment.
  - The model might say that adversarial actions are likely.

For now, assume each chance node magically comes along with the probabilities that specify the distribution over its outcomes.

Informed Probabilities

- Let's say you know that your opponent is sometimes lazy. 20% of the time, she moves randomly, but usually (80%) she runs a depth 2 minimax to decide her move.
- Question: What tree search should you use?

Answer: Expectimax!
- To figure out EACH chance node's probabilities, you have to run a simulation of your opponent.
- This kind of thing gets very slow very quickly.
- Even worse if you have to simulate your opponent simulating you...
- ... except for minimax, which has the nice property that it all collapses into one game tree.

Modeling Assumptions

The Dangers of Optimism and Pessimism

- Dangerous Optimism: Assuming choice when the world is adversarial.
- Dangerous Pessimism: Assuming the worst case when it's not likely.

Video of Demo World Assumptions

Random Ghost – Expectimax Pacman
Video of Demo World Assumptions
Adversarial Ghost – Minimax Pacman

Video of Demo World Assumptions
Adversarial Ghost – Expectimax Pacman

Video of Demo World Assumptions
Random Ghost – Minimax Pacman

Assumptions vs. Reality

<table>
<thead>
<tr>
<th></th>
<th>Minimax Pacman</th>
<th>Expectimax Pacman</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Won 5/5</td>
<td>Won 5/5</td>
</tr>
<tr>
<td></td>
<td>Avg. Score: 493</td>
<td>Avg. Score: 483</td>
</tr>
</tbody>
</table>

Results from playing 5 games

Pacman used depth 4 search with an eval function that avoids trouble
Ghost used depth 2 search with an eval function that seeks Pacman

Other Game Types
Example: Backgammon

Mixed Layer Types

- E.g. Backgammon
- Expectiminimax
  - Environment is an extra "random agent" player that moves after each min/max agent
  - Each node computes the appropriate combination of its children

Example: Backgammon

- Dice rolls increase by 31 possible rolls with 2 dice
  - Backgammon ~ 20 legal moves
  - Depth 2 = 20 x (21 x 20)² = 1.2 x 10⁶
- As depth increases, probability of reaching a given search node shrinks
  - So usefulness of search is diminished
  - So limiting depth is less damaging
- But pruning is trickier...
- Historic AI (1992): TDGammon uses depth-2 search + very good evaluation function + reinforcement learning: world-champion level play
  - 1st AI world champion in any game!

Different Types of Ghosts?

- Stupid
- Devilish Smart

Multi-Agent Utilities

- What if the game is not zero-sum, or has multiple players?
- Generalization of minimax:
  - Terminal have utility tuples
  - Node-values are also utility tuples
  - Each player maximizes its own component
  - Caring for cooperation and competition dynamically...

Utilities

Maximum Expected Utility

- Why should we average utilities?
- Principle of maximum expected utility:
  - A rational agent should chose the action that maximizes its expected utility, given its knowledge
- Questions:
  - Where do utilities come from?
  - How do we know such utilities even exist?
  - How do we know that averaging even makes sense?
  - What if our behavior (preferences) can’t be described by utilities?
What Utilities to Use?

- For worst-case minimax reasoning, terminal function scale doesn’t matter
- We just want better states to have higher evaluations (get the ordering right)
- We call this insensitivity to monotonic transformations
- For average-case expectimax reasoning, we need magnitudes to be meaningful

Utilities

- Utilities are functions from outcomes (states of the world) to real numbers that describe an agent’s preferences
- Where do utilities come from?
  - In a game, may be single (+1/-1)
  - Utilities summarize the agent’s goals
  - Theorem: any “rational” preferences can be summarized as a utility function
- We hard-wire utilities and let behavior emerge
  - Why don’t we let agents pick utilities?
  - Why don’t we prescribe behaviors?

Utilities: Uncertain Outcomes

Preferences

- An agent must have preferences among:
  - Prizes: A, B, etc.
  - Lotteries: situations with uncertain prizes

  \[ \mathcal{L} = \{ p \cdot A, (1-p) \cdot B \} \]

- Notation:
  - Preference: \( A > B \)
  - Indifference: \( A \sim B \)

Preferences

- An agent must have preferences among:
  - Prizes: A, B, etc.
  - Lotteries: situations with uncertain prizes

  \[ \mathcal{L} = \{ p \cdot A, (1-p) \cdot B \} \]

- Notation:
  - Preference: \( A > B \)
  - Indifference: \( A \sim B \)

Rationality

- We want some constraints on preferences before we call them rational, such as:

  - Axiom of Transitivity:
    \[ (A \succ B) \cap (B \succ C) \Rightarrow (A \succ C) \]

  - For example: an agent with intransitive preferences can be induced to give away all of its money
    - If B > C, then an agent with C would pay (say) 1 cent to get B
    - If A > B, then an agent with B would pay (say) 1 cent to get A
    - If C > A, then an agent with A would pay (say) 1 cent to get C
**Rational Preferences**

The Axioms of Rationality

- **Unanimity:** (A > B) ∧ (B > C) ⇒ (A > C)
- **Transitivity:** (A > B) ∧ (B > C) ⇒ (A > C)
- **Completeness:** A = B ∨ A > B ∨ B > A
- **Continuity:** A > B ⇔ [a, p, A, 1 − p, C] > B
- **Monotonicity:** [a, p, A, 1 − p, C] > [a, p, a, 1 − p, C]

Theorem: Rational preferences imply behavior describable as maximization of expected utility

**MEU Principle**

- **Theorem** (Ramsey, 1931; von Neumann & Morgenstern, 1944)
  - Given any preferences satisfying these constraints, there exists a real-valued function U such that:
    
    $$U(A) ≥ U(B) ⇔ A ≥ B$$
    
    $$U(p_1, S_1; \ldots; p_n, S_n) = \sum p_i U(S_i)$$

  - I.e., values assigned by U preserve preferences of both prizes and lotteries

  - Maximum expected utility (MEU) principle:
    - Choose the action that maximizes expected utility
    - Note: an agent can be entirely rational (consistent with MEU) without ever representing or manipulating utilities and probabilities
    - E.g., a lookup table for perfect tic-tac-toe, a reflex vacuum cleaner

**Human Utilities**

- **Utilities map states to real numbers. Which numbers?**
- **Standard approach to assessment (elicitation) of human utilities:**
  - Compare a prize A to a standard lottery L between
    - "best possible prize" u+ with probability p
    - "worst possible catastrophe" u- with probability 1-p
  - Adjust lottery probability p until indifference: A ~ L
  - Resulting p is a utility in [0,1]

**Utility Scales**

- **Normalized utilities:** u+ = 1.0, u- = 0.0
- **Micromorts:** one-millionth chance of death, useful for paying to reduce product risks, etc.
- **QALYs:** quality-adjusted life years, useful for medical decisions involving substantial risk
  - Note: behavior is invariant under positive linear transformation
    
    $$U'(x) = k_1 U(x) + k_2$$
    
    where k1 > 0

  - With deterministic prizes only (no lottery choices), only ordinal utility can be determined, i.e., total order on prizes

**Human Utilities**

- **Money does not** behave as a utility function, but we can talk about the utility of having money (or being in debt)
- **Given a lottery L = [p, S; (1-p), D]**
  - **The expected monetary value (EMV)** is p*S + (1-p)*D
  - **Typically, EMV(L) = £100, £500**
  - In this sense, people are risk-averse
  - When deep in debt, people are risk-seeking

**Money**

- **Pay $30**
  - $0.999999
  - $0.000001
  - No change
  - Instant death
Example: Insurance

- Consider the lottery \([0.5, \$1000; 0.5, \$0]\)
  - What is its expected monetary value? \(\$500\)
  - What is its certainty equivalent?
    - Monetary value acceptable in lieu of lottery
    - \(\$400\) for most people
  - Difference of \(\$100\) is the insurance premium
    - There’s an insurance industry because people
      will pay to reduce their risk
    - If everyone were risk-neutral, no insurance
      needed
    - It’s unethical; you’d rather have the \$400 and
      the insurance company would rather have
      the lottery (their utility curve is flat and they
      have many lotteries)

Example: Human Rationality?

- Famous example of Allais (1953)
  - A: \([0.8, \$4k; 0.2, \$0]\)
  - B: \([1.0, \$3k; 0.0, \$0]\)
  - C: \([0.2, \$4k; 0.8, \$0]\)
  - D: \([0.25, \$3k; 0.75, \$0]\)
- Most people prefer B > A, C > D
- But if \(U(\$0) = 0\), then
  - B > A \(\Rightarrow U(\$3k) > 0.8 U(\$4k)\)
  - C > D \(\Rightarrow 0.2 U(\$4k) > U(\$3k)\)

Kahneman & Tversky

Choose between

<table>
<thead>
<tr>
<th>Option A</th>
<th>Option B</th>
</tr>
</thead>
<tbody>
<tr>
<td>33%</td>
<td>100%</td>
</tr>
<tr>
<td>$2500</td>
<td>$2400</td>
</tr>
<tr>
<td>66%</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>[18]</td>
</tr>
<tr>
<td>01%</td>
<td>[82]*</td>
</tr>
</tbody>
</table>

Kahneman & Tversky

Choose between

<table>
<thead>
<tr>
<th>Option C</th>
<th>Option D</th>
</tr>
</thead>
<tbody>
<tr>
<td>33%</td>
<td>34%</td>
</tr>
<tr>
<td>$2500</td>
<td>$2400</td>
</tr>
<tr>
<td>67%</td>
<td>66%</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>[83]*</td>
<td>[17]</td>
</tr>
</tbody>
</table>

Kahneman & Tversky

Choose between

<table>
<thead>
<tr>
<th>Option A</th>
<th>Option B</th>
</tr>
</thead>
<tbody>
<tr>
<td>33%</td>
<td>100%</td>
</tr>
<tr>
<td>$2500</td>
<td>$2400</td>
</tr>
<tr>
<td>66%</td>
<td>0</td>
</tr>
<tr>
<td>01%</td>
<td>[18]</td>
</tr>
<tr>
<td>0</td>
<td>[82]*</td>
</tr>
</tbody>
</table>

Kahneman & Tversky

Choose between

<table>
<thead>
<tr>
<th>Option C</th>
<th>Option D</th>
</tr>
</thead>
<tbody>
<tr>
<td>33%</td>
<td>34%</td>
</tr>
<tr>
<td>$2500</td>
<td>$2400</td>
</tr>
<tr>
<td>67%</td>
<td>66%</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>[83]*</td>
<td>[17]</td>
</tr>
</tbody>
</table>