CSE 473: Artificial Intelligence

Autumn 2015

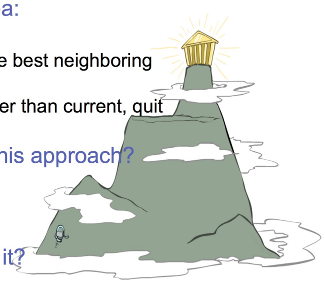
Hill Climbing
Expectimax Search
Uncertainty

Fereshteh Sadeghi and Steve Tanimoto

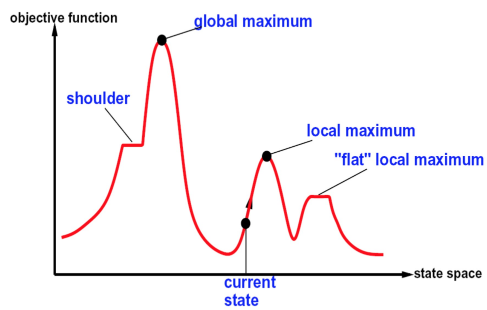
With slides from :
Dieter Fox, Dan Weld, Dan Klein, Pieter Abbeel and others.

Hill Climbing

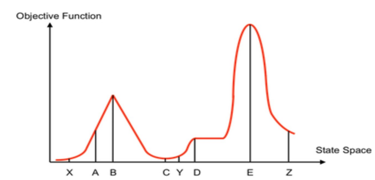
- Simple, general idea:
  - Start wherever
  - Repeat: move to the best neighboring state
  - If no neighbors better than current, quit
- What's bad about this approach?
  - Complete?
  - Optimal?
- What's good about it?



Hill Climbing Diagram



Hill Climbing



Starting from X, where do you end up ?

Starting from Y, where do you end up ?

Starting from Z, where do you end up ?

Simulated Annealing

- Idea: Escape local maxima by allowing downhill moves
  - But make them rarer as time goes on

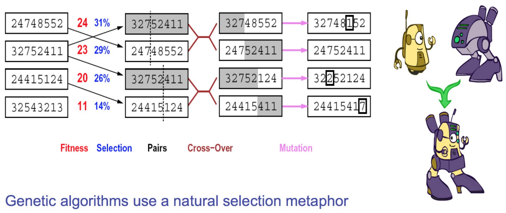
```
function SIMULATED-ANNEALING( problem, schedule) returns a solution state
   inputs: problem, a problem
           schedule, a mapping from time to "temperature"
   local variables: current, a node
                    next, a node
                    T, a "temperature" controlling prob. of downward steps

   current ← MAKE-NODE(INITIAL-STATE[problem])
   for t ← 1 to ∞ do
      T ← schedule[t]
      if T = 0 then return current
      next ← a randomly selected successor of current
      ΔE ← VALUE[next] – VALUE[current]
      if ΔE > 0 then current ← next
      else current ← next only with probability e^(Δ E/T)
```

Simulated Annealing

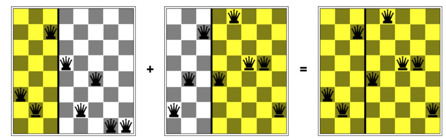
- Theoretical guarantee:
  - Stationary distribution: $p(x) \propto e^{\frac{E(x)}{kT}}$
  - If T decreased slowly enough, will converge to optimal state!
- Is this an interesting guarantee?
- Sounds like magic, but reality is reality:
  - The more downhill steps you need to escape a local optimum, the less likely you are to ever make them all in a row
  - People think hard about *ridge operators* which let you jump around the space in better ways

Genetic Algorithms



- Genetic algorithms use a natural selection metaphor
  - Keep best N hypotheses at each step (selection) based on a fitness function
  - Also have pairwise crossover operators, with optional mutation to give variety
- Possibly the most misunderstood, misapplied (and even maligned) technique around

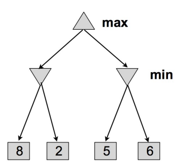
Example: N-Queens



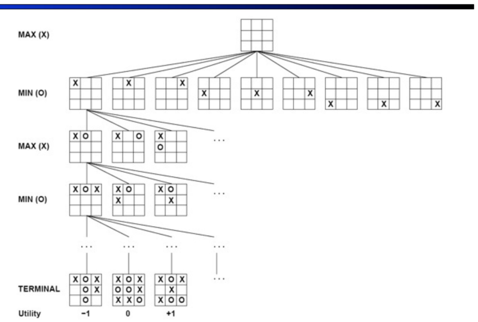
- Why does crossover make sense here?
- When wouldn't it make sense?
- What would mutation be?

Deterministic Two-Player

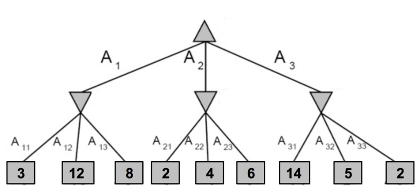
- E.g. tic-tac-toe, chess, checkers
- Zero-sum games
  - One player maximizes result
  - The other minimizes result
- **Minimax search**
  - A state-space search tree
  - Players alternate
  - Choose move to position with highest minimax value = best achievable utility against best play



Tic-tac-toe Game Tree



Minimax Example



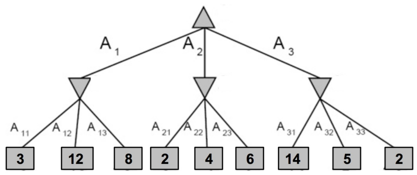
Minimax Implementation

```
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 MIN: return min-value(state)
```

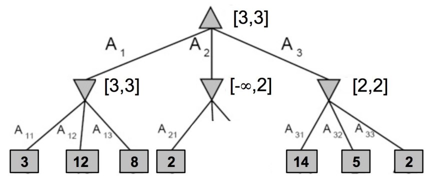
```
def max-value(state):
    initialize v = -∞
    for each successor of state:
        v = max(v, min-value(successor))
    return v
```

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

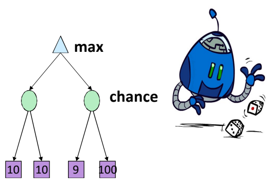
Can we do better?



α-β Pruning Example

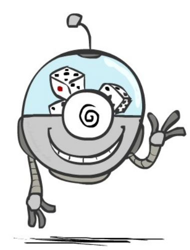


Worst-Case vs. Average Case



Idea: Uncertain outcomes controlled by chance!

Probabilities

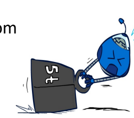


Reminder: Probabilities

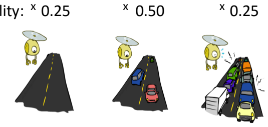
- A random variable represents an event whose outcome is unknown
- A probability distribution is an assignment of weights to outcomes
- Example: Traffic on freeway
  - Random variable: T = whether there's traffic
  - Outcomes: T in {none, light, heavy}
  - Distribution: P(T=none) = 0.25, P(T=light) = 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 | Hour=8am) = 0.60
  - We'll talk about methods for reasoning and updating probabilities later

Reminder: Expectations

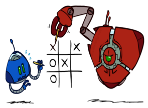
- The expected value of a function of a random variable is the average, weighted by the probability distribution over outcomes
- Example: How long to get to the airport?

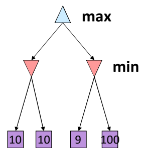


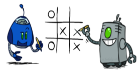
Time:	20 min	+	30 min	+	60 min	⇨	35 min
Probability:	x 0.25		x 0.50		x 0.25		



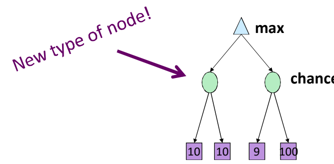
Worst-Case vs. Average Case







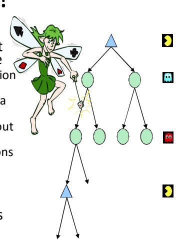
Worst-Case vs. Average Case



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

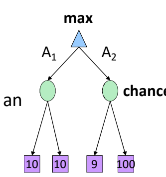
What Probabilities to Use?

- 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: opponent or environment
  - The model might say that adversarial actions are likely!
- For now, assume each chance node ***magically*** comes along with probabilities that specify the distribution over its outcomes



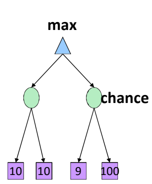
Randomness?

- Why wouldn't we know the results of an action?
  - Explicit randomness: rolling dice
  - Unpredictable opponents: the ghosts respond erratically
  - Actions can fail: when robot moves, its wheels might slip



Expectimax Search

- Values now reflect average-case (expected) outcomes, not worst-case (minimum) outcomes
- Expectimax search: Compute 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



[Demo: min vs exp (L7D1,2)]

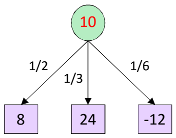
Expectimax Pseudocode

```
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
```

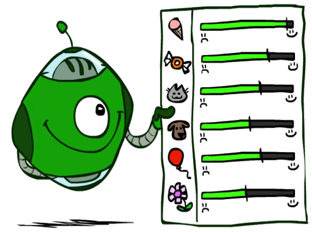
Expectimax Pseudocode



v = (1/2) (8) + (1/3) (24) + (1/6) (-12) = 10

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

Utilities

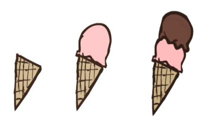


Maximum Expected Utility

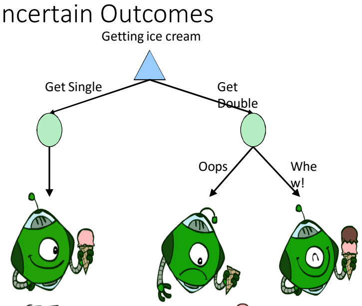
- Why should we average utilities?
- Principle of maximum expected utility:
  - A rational agent should chose the action that maximizes its expected utility, given its knowledge
- Questions:
  - Where do utilities come from?
  - How do we know such utilities even exist?
  - How do we know that averaging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

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 simple (+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 behaviors emerge
  - Why don't we let agents pick utilities?
  - Why don't we prescribe behaviors?



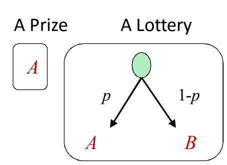
Utilities: Uncertain Outcomes



Preferences

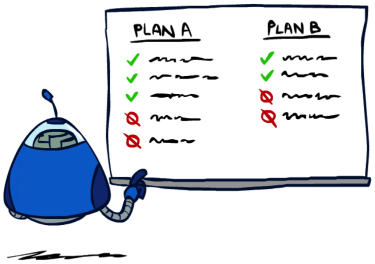
- An agent must have preferences among:
  - Prizes: A, B, etc.
  - Lotteries: situations with uncertain prizes

$$L = [p, A;\ (1-p), B]$$



- Notation:
  - Preference: $A \succ B$
  - Indifference: $A \sim B$

Rationality

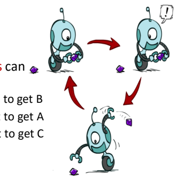


Rational Preferences

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

Axiom of Transitivity: $(A \succ B) \wedge (B \succ C) \Rightarrow (A \succ C)$

- 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

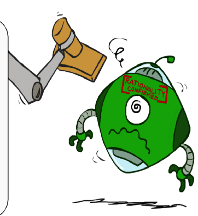
Orderability
$(A \succ B) \vee (B \succ A) \vee (A \sim B)$

Transitivity
$(A \succ B) \wedge (B \succ C) \Rightarrow (A \succ C)$

Continuity
$A \succ B \succ C \Rightarrow \exists p\ [p, A;\ 1-p, C] \sim B$

Substitutability
$A \sim B \Rightarrow [p, A;\ 1-p, C] \sim [p, B; 1-p, C]$

Monotonicity
$A \succ B \Rightarrow$
$(p \geq q \Leftrightarrow [p, A;\ 1-p, B] \succeq [q, A;\ 1-q, B])$



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

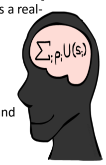
MEU Principle

- Theorem [Ramsey, 1931; von Neumann & Morgenstern, 1944]
  - Given any preferences satisfying these constraints, there exists a real-valued function U such that:

$$U(A) \geq U(B) \ \Leftrightarrow\ A \succeq B$$

$$U([p_1, S_1;\ \ldots\ ;\ p_n, S_n]) = \sum_i p_i U(S_i)$$

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

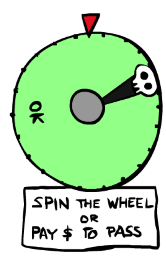


- Maximum expected utility (MEU) principle:
  - Choose the action that maximizes expected utility
  - Note: an agent can be entirely rational (consistent with MEU) without ever representing or manipulating utilities and probabilities
  - E.g., a lookup table for perfect tic-tac-toe, a reflex vacuum cleaner

Human Utilities

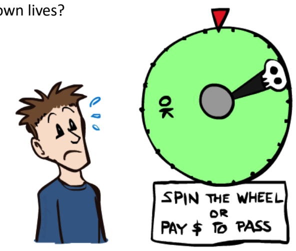
Human Utilities

Playing Russian Roulette?



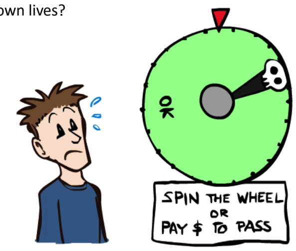
Playing Russian Roulette?

How much you would pay to avoid a a risk?
What value people would place on their own lives?



Playing Russian Roulette?

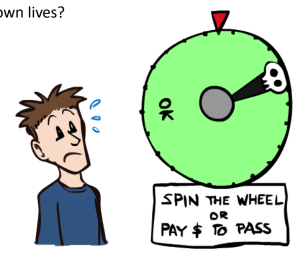
How much you would pay to avoid a a risk?
What value people would place on their own lives?
Perhaps ***tens of thousands of dollars***…??



Playing Russian Roulette?

How much you would pay to avoid a a risk?
What value people would place on their own lives?
Perhaps ***tens of thousands of dollars***…??

➡ **micromort**

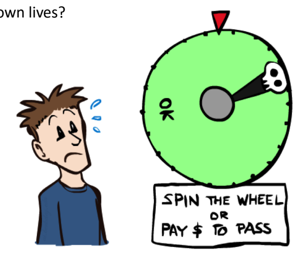


Playing Russian Roulette?

How much you would pay to avoid a a risk?
What value people would place on their own lives?
Perhaps ***tens of thousands of dollars***…??

➡ **micromort**

The actual human behavior reflects a much lower monetary value for a micromort!!!



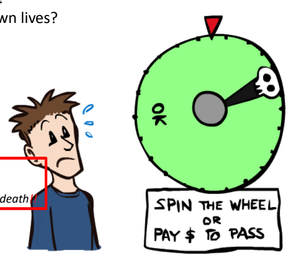
Playing Russian Roulette?

How much you would pay to avoid a a risk?
What value people would place on their own lives?
Perhaps ***tens of thousands of dollars***…??

➡ **micromort**

The actual human behavior reflects a much lower monetary value for a micromort!!!

Driving for 230 miles incurs a risk of one micromort!!
Over the life of your car (~92k miles) that's 400 micromorts!!
People are willing to pay $10k for a car that halves the risk of death!!



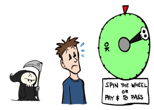
Utility Scales

- 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

$$U'(x) = k_1 U(x) + k_2 \quad \text{where } k_1 > 0$$

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

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_p between
    - "best possible prize" u_+ with probability p
    - "worst possible catastrophe" u_- with probability 1-p
  - Adjust lottery probability p until indifference: $A \sim L_p$
  - Resulting p is a utility in [0,1]

Pay $30 ~ [0.999999 → *No change* ; 0.000001 → *Instant death*]

Utility of Money

- Money plays a significant rule in human utility functions
- Usually an agent prefers more money to less

Utility of Money

- Money plays a significant rule in human utility functions
- Usually an agent prefers more money to less

⇨ The agent exhibits a monotonic preference for more money

Utility of Money

- Money plays a significant rule in human utility functions
- Usually an agent prefers more money to less

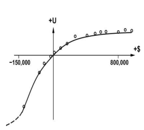
⇨ The agent exhibits a monotonic preference for more money

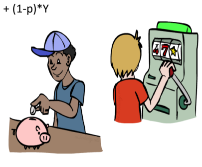
But!

- ***This does not mean that money behaves as a utility function!***
- ***This does not say anything about preferences between lotteries involving money!***

Money

- 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, $X; (1-p), $Y]
  - The expected monetary value EMV(L) is p*X + (1-p)*Y
  - U(L) = p*U($X) + (1-p)*U($Y)
  - Typically, U(L) < U(EMV(L))
  - In this sense, people are risk-averse
  - When deep in debt, people are risk-prone





Example:

- In a television game show:

A) take $1,000,000 prize

B) gamble on the flip of a coin:

- If heads nothing
- If tails get $2,500,000

Which one you would take? A or B?

Example:

- In a television game show:

A) take $1,000,000 prize

B) gamble on the flip of a coin:

  - If heads nothing
  - If tails get $2,500,000

- If coin is fair, Expected Monetary Value (EMV) of gamble is:

EMV = ½ ($0) + ½ ($2,500,000) = $1,250,000

→ more than $1,000000

Example:

- In a television game show:

A) take $1,000,000 prize

B) gamble on the flip of a coin:

  - If heads nothing
  - If tails get $2,500,000

- If coin is fair, Expected Monetary Value (EMV) of gamble is:

EMV = ½ ($0) + ½ ($2,500,000) = $1,250,000

→ more than $1,000000

Would you choose B?

Example:

- In a television game show:

A) take $1,000,000 prize

B) gamble on the flip of a coin:

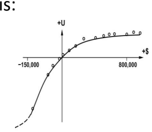
  - If heads nothing
  - If tails get $2,500,000

- If coin is fair, Expected Monetary Value (EMV) of gamble is:

EMV = ½ ($0) + ½ ($2,500,000) = $1,250,000

EU(B) = ½ U(Sk) + ½ U(Sk + $2,500,000)

EU(A) = U(Sk + $1,000,000)



Example:

EMV = ½ ($0) + ½ ($2,500,000) = $1,250,000

EU(B) = ½ U(Sk) + ½ U(Sk + $2,500,000)

EU(A) = U(Sk + $1,000,000)

Utility is not directly proportional to monetary value

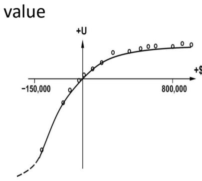
Utility(first million) is very high!

Utility(additional million) is smaller!

U(Sk) = 5,

U(Sk + $1,000,000) = 8

U(Sk + $2,500,000) = 9



Example:

EMV = ½ ($0) + ½ ($2,500,000) = $1,250,000

EU(B) = ½ U(Sk) + ½ U(Sk + $2,500,000) **= 7**

EU(A) = U(Sk + $1,000,000) **= 8**

Utility is not directly proportional to monetary value

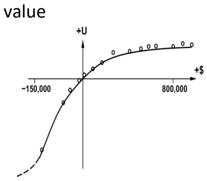
Utility(first million) is very high!

Utility(additional million) is smaller!

U(Sk) = 5,

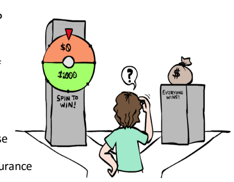
U(Sk + $1,000,000) = 8

U(Sk + $2,500,000) = 9



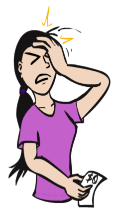
Example: Insurance

- 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 win-win: 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)

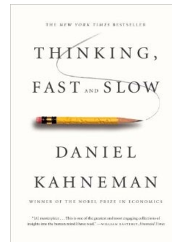


Example: Human Rationality?

- 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 ⇒ U($3k) > 0.8 U($4k)
  - C > D ⇒ 0.8 U($4k) > U($3k)



Recommended



- Risks vs gains
- Probability estimates
- Cognitive architecture
- Much more

56