**2/2/18**

11-MDPs

Each MDP state projects an expectimax-like search tree

**Utilities of Sequences**

* **Different kinds of sequences**
  + sequence of rewards matter? Does order matter? Less or more?
  + getting rewards one at a time or all at the end?
* What preferences should an agent have over reward sequences?
  + [1,2,2] or [2,3,4]
  + [0,0,1] or [1,0,0]

**Discounting**

* reasonable to maximise sum of rewards
* reasonable to prefer rewards now than to rewards later.
* one solution: value of rewards decay exponentially
* **How:**
  + each time we descend a level in a tree, multiply the discount once
* **Why?**
  + Sooner rewards probably do have higher utility than later rewards
  + Also helps our algorithms converge

**Stationary Preferences**

* Suppose we have 2 reward sequences {} and {}
* two ways to define utilities:
  + additive utility:
  + discounted utility: U({a\_n}) =
* if preferences are unchanged after applying utilities, then preferences are stationary

**Infinite Utilities**

* Problem: game lasts forever? Infinite rewards?
* Solutions:
  + finite horizons: (similar to depth limited search)
    - what if game ended in 2 turns
    - terminate episodes after a fixed T steps (e.g. life)
    - Gives non stationary policies (depends on time left)
  + Discounting: Use
    - smaller means smaller horizon - shorter term focus
  + **Absorbing state:** guarantee that for every policy, a terminal state will eventually be reached (like “overheated” for racing)