Lecture4 Game Playing

- Type: Turn-Based(Main Focused on) & Real-Time & Simultaneous
- Task
  - Given
    - S start
    - IsEnd(s)
    - Action(s)
    - Utility(s): Reward for the end state s
    - Succ(s,a) = s'
    - Players
    - Player(s) ∈Players: Which player moves in s
  - Output: Path (Sequence of actions)
- Example: Bin Selection
  - Goal: Maximize the chosen number
  - Opponent
    - Adversarial
    - Helpful
    - Stochastic
- Node Type
  - Node: A state and policy(Π) of a player
  - Game
    - Min Node: Downward-pointing triangle ▽
    - Max Node: Upward-pointing triangle △
    - Chance Node: Cicle Probability∈(0,1) ○
- Value Function
  - V(s) denotes the value (utility) of state s　计算方式类似求期望
  - $$V(s,\mathbf{d})=\begin{cases}\textbf{Utility} & \textbf{isEnd(s)}\\ \textbf{Eval(s)} & \textbf{d=0}\\ \sum_{a\in Actions(s)} V(Succ(s,\pi_{agent}(s)),\ \mathbf{d-1}) & Player(s)=agent\\ \sum_{a\in Actions(s)} V(Succ(s,\pi_{opp}(s)),\ \mathbf{d-1}) & Player(s)=opp\end{cases}$$
- Minimax
  - Zero-Sum Game: Other name: MinMax, MM, Saddle Point
  - Assumes opponent selects the worst action to an agent
- Time Complexity
- Advanced Methods
  - Evaluation Function
    - Approximate the utility [Eval(s)]
    - Can stop earlier
    - Require domain-specific knowledge
  - Alpha-Beta Pruning
    - Basic
      - Compute TRUE utility
      - Ignore unnecessary path
      - General-Purpose
    - Algorithm
      - α下限初始-∞，β上限初始+∞
      - max层改变α，α = max {这层α，下层α，β}
      - min层改变β，β = min {这层β，下层α，β}
      - 顺序：先左子树，再返回父节点，再右子树
      - 一旦出现α ≥ β，即可对右侧分支进行剪枝操作
- Simultaneous Game
  - Example
    - Two-Finger Morra
    - Prisoner's Dilemma