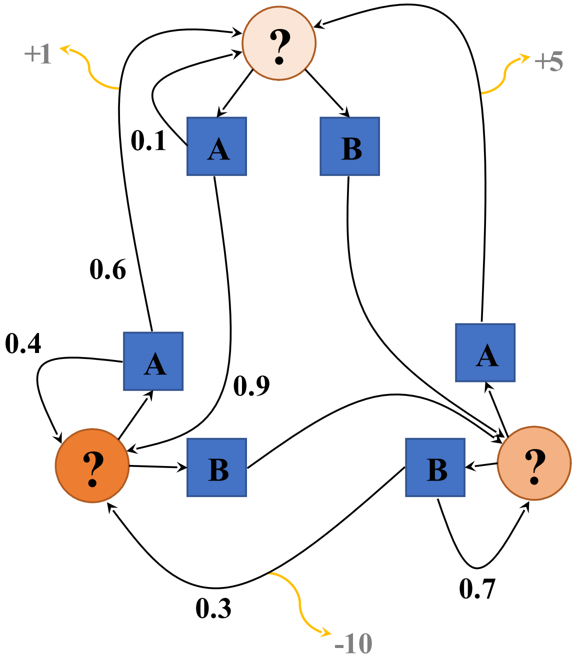
Incomplete Information Dynamic Games

Incomplete Information

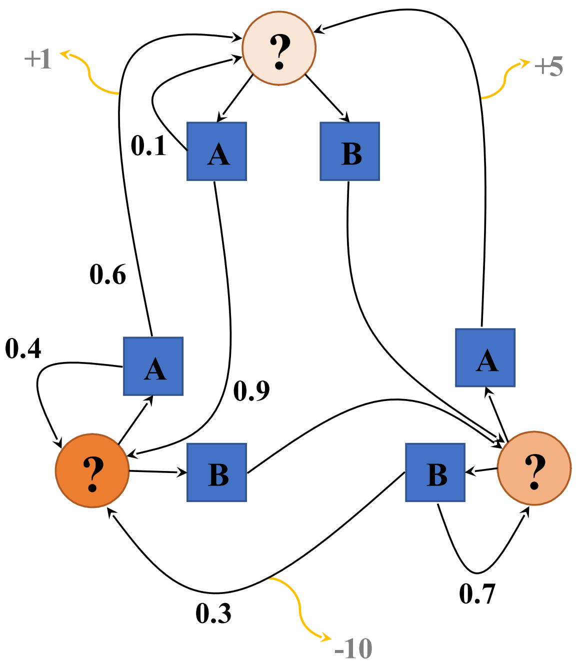
Partially Observable Markov Decision Process (POMDP)



- $\mathcal{S}$ - State space
- $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$ - Transition probability distribution
- $\mathcal{A}$ - Action space
- $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ - Reward

Alleatory

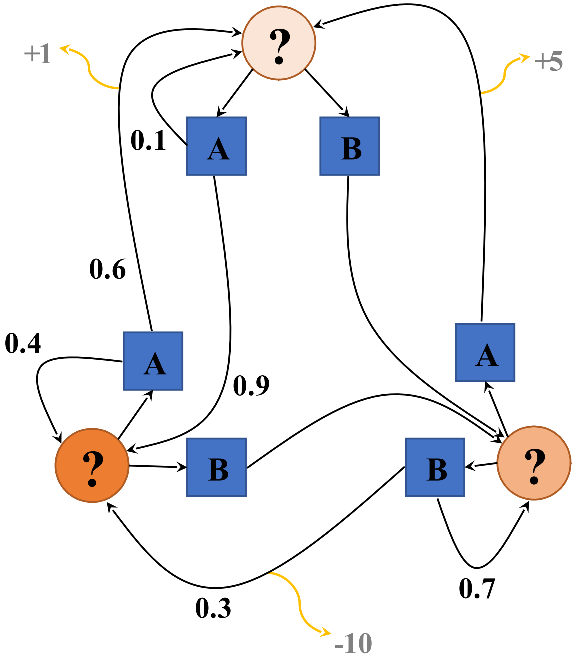
Partially Observable Markov Decision Process (POMDP)



- $\mathcal{S}$ - State space
- $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ - Transition probability distribution
- $\mathcal{A}$ - Action space
- $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ - Reward
- $\mathcal{O}$ - Observation space

Alleatory

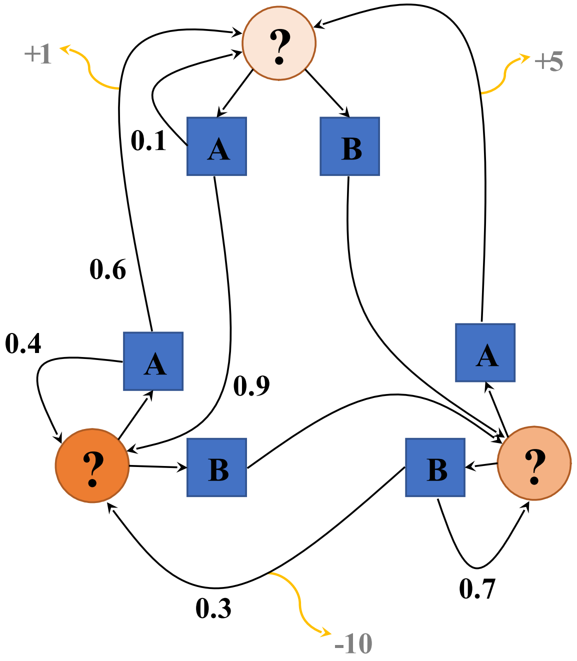
Partially Observable Markov Decision Process (POMDP)



- $\mathcal{S}$ - State space
- $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathbb{R}$ - Transition probability distribution
- $\mathcal{A}$ - Action space
- $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ - Reward
- $\mathcal{O}$ - Observation space
- $Z : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \times \mathcal{O} \rightarrow \mathbb{R}$ - Observation probability distribution

Alleatory

Partially Observable Markov Decision Process (POMDP)



- $\mathcal{S}$ - State space
- $T : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ - Transition probability distribution
- $\mathcal{A}$ - Action space
- $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ - Reward
- $\mathcal{O}$ - Observation space
- $Z : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \times \mathcal{O} \to \mathbb{R}$ - Observation probability distribution

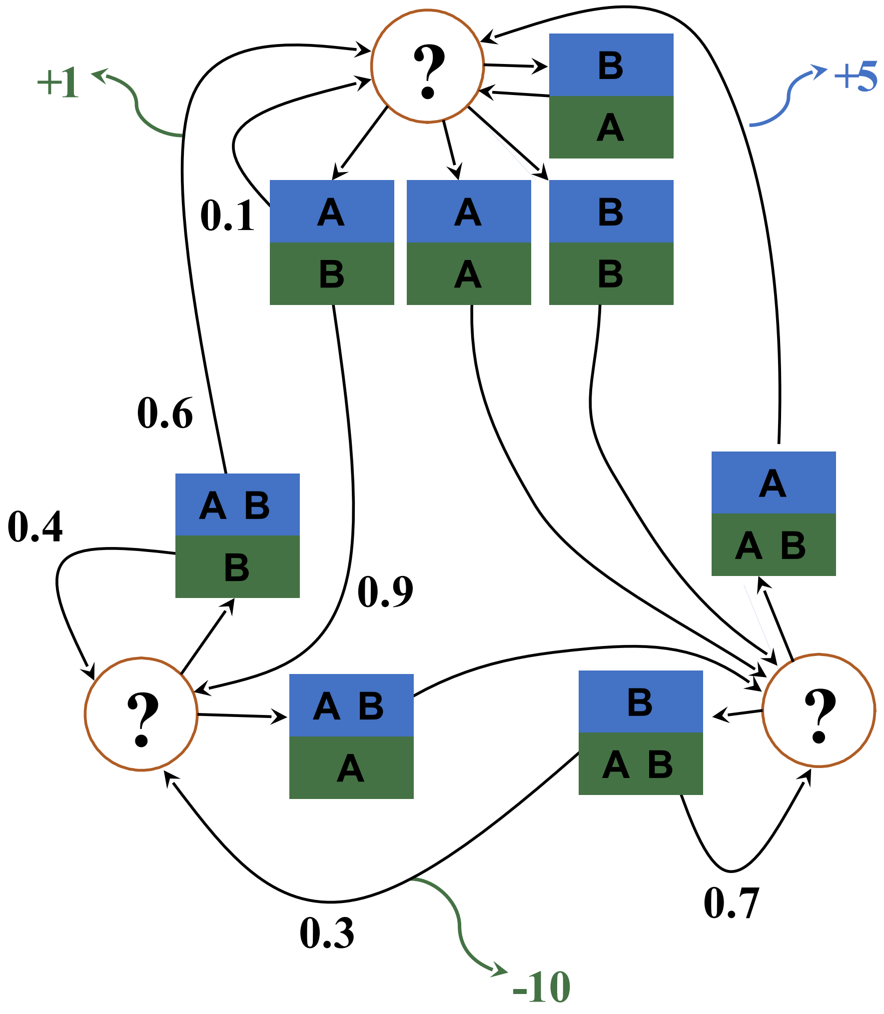
Alleatory

Epistemic (Static)

Epistemic (Dynamic)

Partially Observable Markov Game

$(S, T, \{A^i\}, \{R^i\}, \{O^i\}, \{Z^i\}, \gamma)$



- $\mathcal{S}$ - State space
- $T(s' \mid s, \boldsymbol{a})$ - Transition probability distribution
  Joint actions
- $\mathcal{A}^i$, $i \in 1..k$ - Action spaces
- $R^i(s, \boldsymbol{a})$ - Reward function
- $\mathcal{O}^i$, $i \in 1..k$ - Observation space
- $Z(o^i \mid \boldsymbol{a}, s')$ - Observation probability distribution

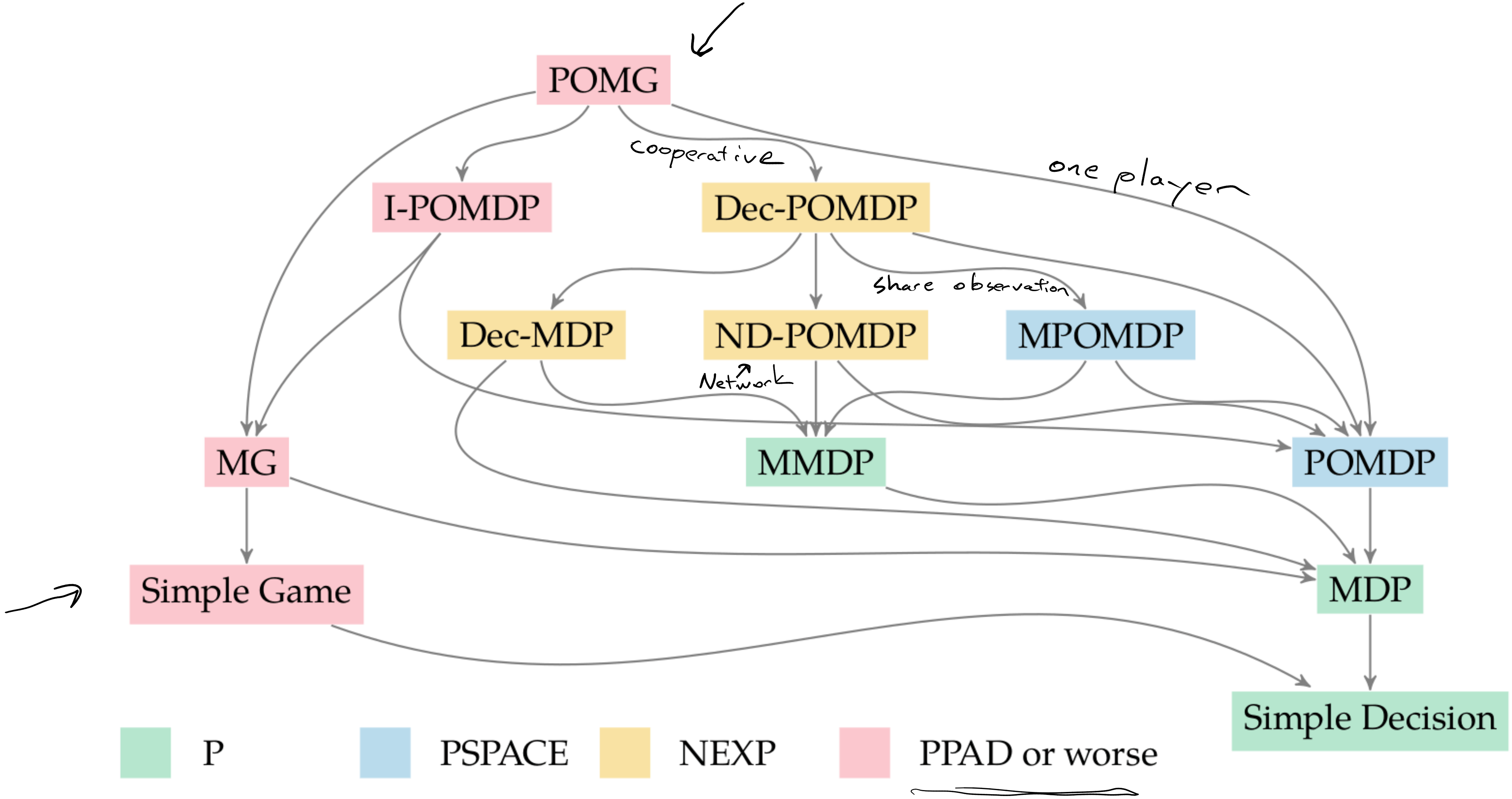
Alleatory

Epistemic (Static)

Epistemic (Dynamic)

Interaction

Hierarchy of Problems



Belief updates?

POMDP: $b'(s') \propto Z(o \mid a, s') \sum_s T(s' \mid s, a)\, b(s)$

POMG $Z(o \mid \vec{a}, s') \sum_s T(s' \mid s, \vec{a})\, b(s)$ X

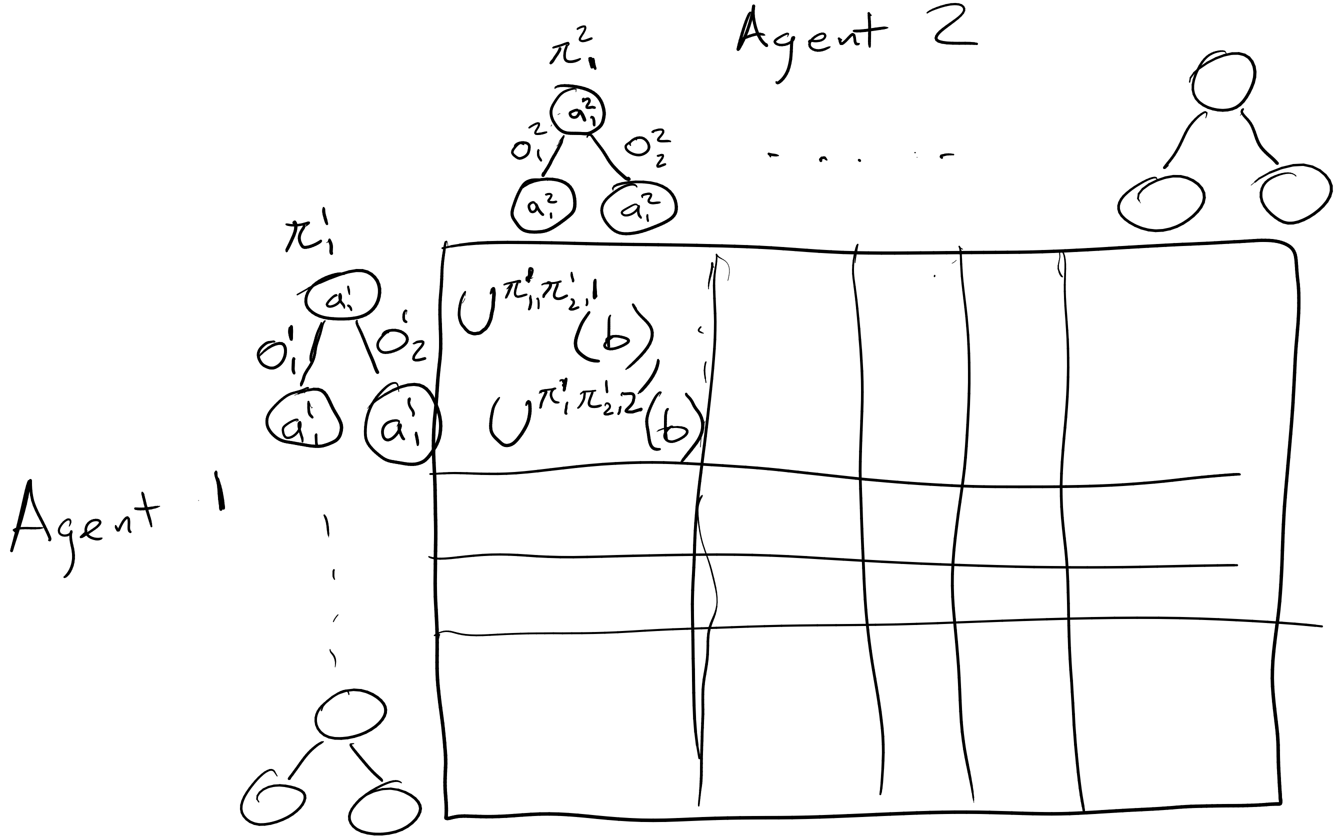
joint action

joint actions

π^{-i}

Problem: Usually trying to solve for π^{-i} at the same time as choosing our actions

Reduction to Simple Game



2 ways

1. Dynamic Programming with Pruning
2. Best responses "Double Oracle"
   - start with strategy
   - compute best response
   - add best response to matrix game
   - solve matrix game

Pruning in Dynamic Programming

Start with all possible N=1-step policies

Loop

  Evaluate N+1 step policies

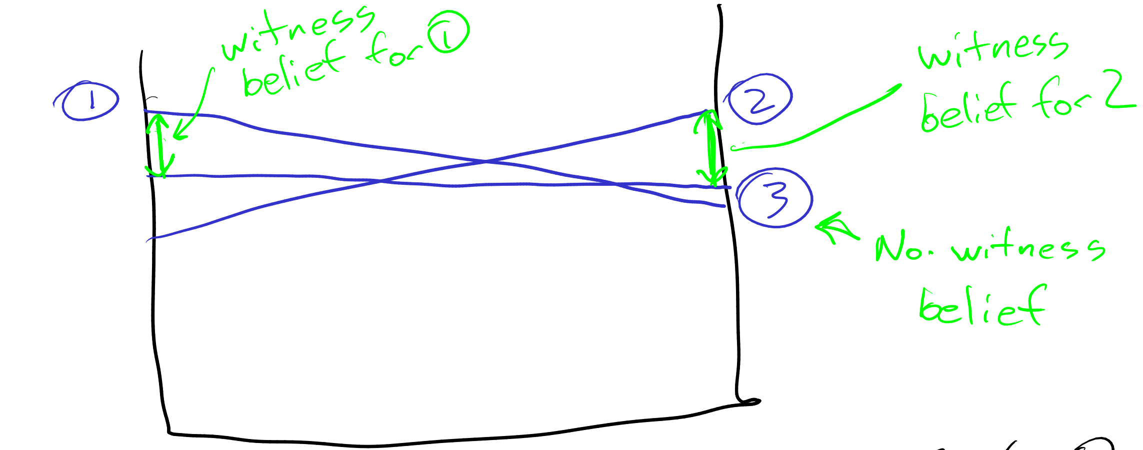
  Prune dominated policies

If there exists $\pi^{i'}$ such that

$$\sum_{\pi^{-i}}\sum_{s} b(\pi^{-i},s)\, U^{\pi^{i'},\pi^{-i}_i}(s) \geq \sum_{\pi^{i}}\sum_{s} b(\pi^{-i},s)\, U^{\pi^{i},\pi^{-i}_i}(s)$$

for all beliefs, we can prune π^i.

POMDPs



$$\underset{\delta, b}{\text{maximize}} \quad \delta$$

subject to $b(\pi^{-i},s) \geq 0 \quad \forall \pi^{-i}, s$

$$\sum_{\pi^{-i}}\sum_{s} b(\pi^{-i},s) = 1$$

$$\sum_{\pi^{-i}}\sum_{s} b(\pi^{-i},s)\left(U^{\pi^{i'},\pi^{-i}_i}(s) - U^{\pi^{i},\pi^{-i}_i}(s)\right) \geq \delta \quad \forall \pi^{i'}$$

If $\delta > 0$ keep π^i

Extensive Form Game

(Alternative to POMGs that is more common in the literature)

- Similar to a minimax tree for a turn-taking game
- Chance nodes
- Information sets

King-Ace Poker Example

King-Ace Poker Example

- 4 Cards: 2 Aces, 2 Kings

King-Ace Poker Example

- 4 Cards: 2 Aces, 2 Kings
- Each player is dealt a card

King-Ace Poker Example

- 4 Cards: 2 Aces, 2 Kings
- Each player is dealt a card
- P1 can either *raise* (r) the payoff to 2 points or *check* (k) the payoff at 1 point

King-Ace Poker Example

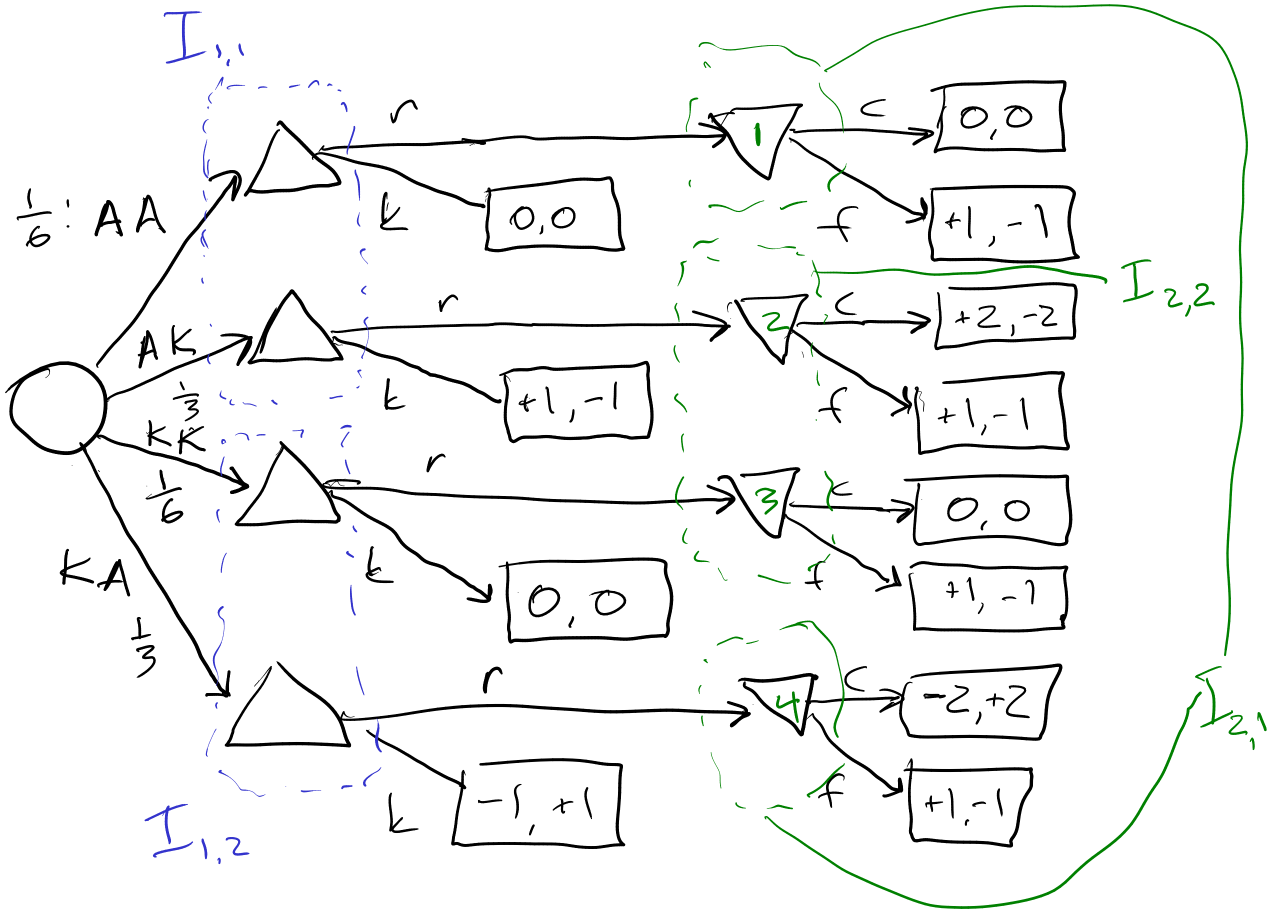
- 4 Cards: 2 Aces, 2 Kings
- Each player is dealt a card
- P1 can either *raise* (r) the payoff to 2 points or *check* (k) the payoff at 1 point
- If P1 raises, P2 can either *call* (c) Player 1's bet, or *fold* (f) the payoff back to 1 point

King-Ace Poker Example

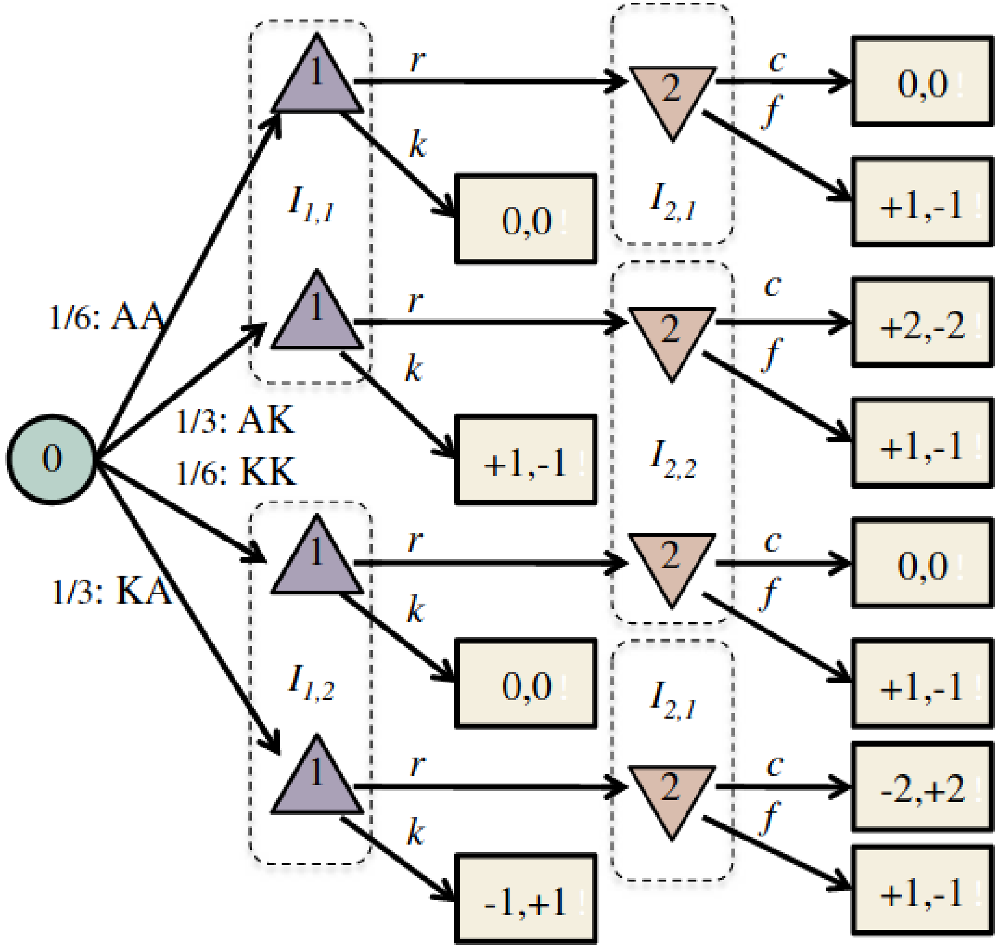
r = raise
k = check

c = call
f = fold

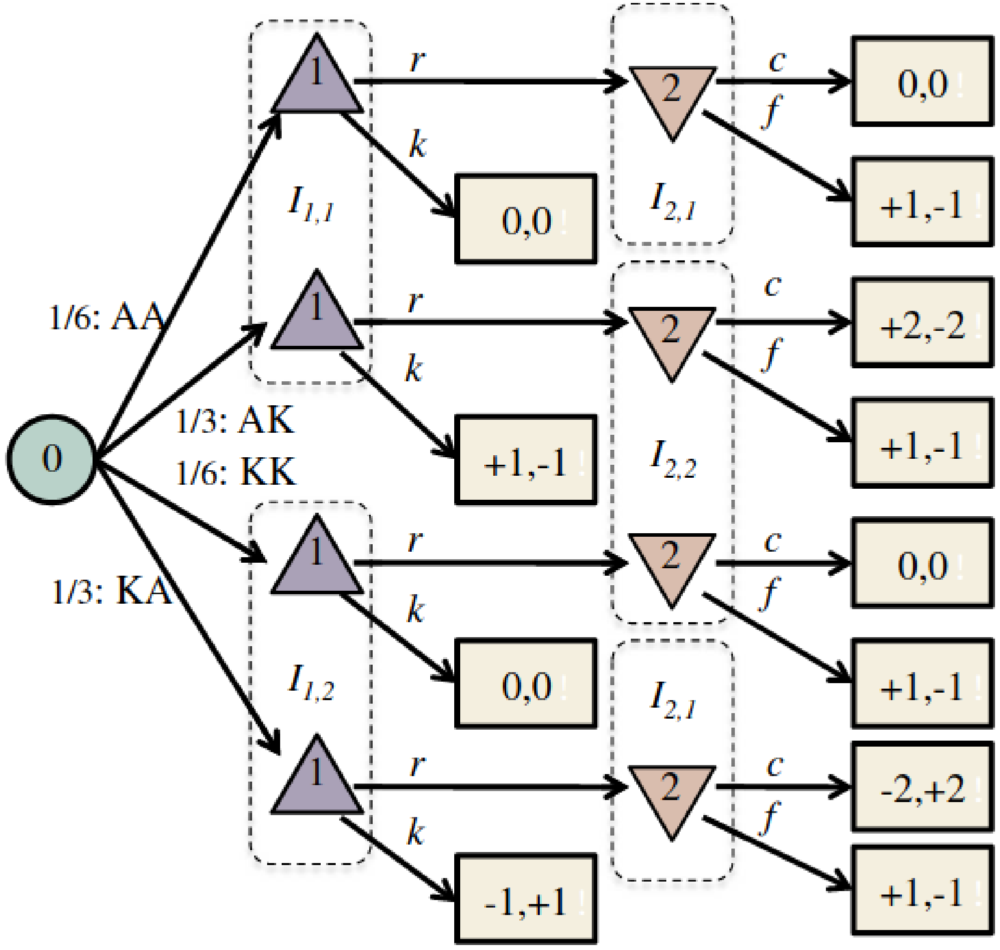
- 4 Cards: 2 Aces, 2 Kings
- Each player is dealt a card
- P1 can either *raise* (r) the payoff to 2 points or *check* (k) the payoff at 1 point
- If P1 raises, P2 can either *call* (c) Player 1's bet, or *fold* (f) the payoff back to 1 point
- The highest card wins



Extensive to Matrix Form

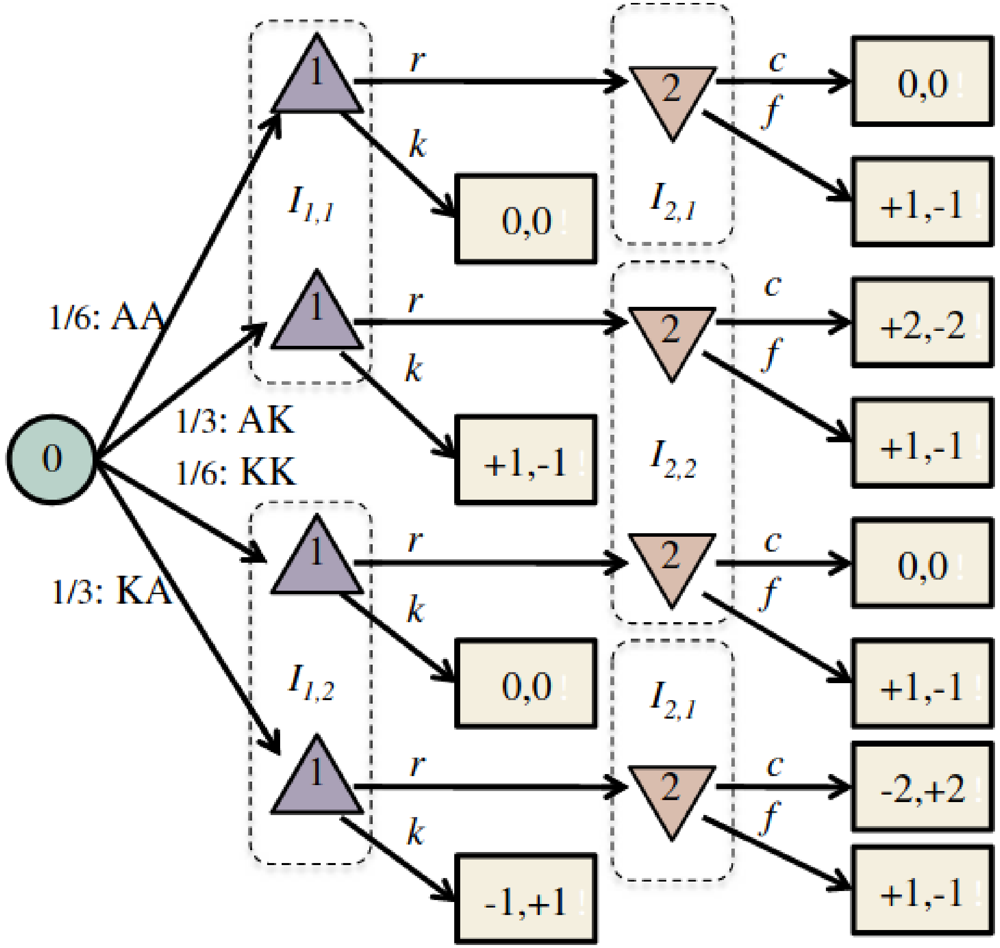


Extensive to Matrix Form



AK	AK 2:cc	2:cf	2:ff	2:fc
1:rr	0	-1/6	1	7/6
1:kr	-1/3	-1/6	5/6	2/3
1:rk	1/3	0	1/6	1/2
1:kk	0	0	0	0

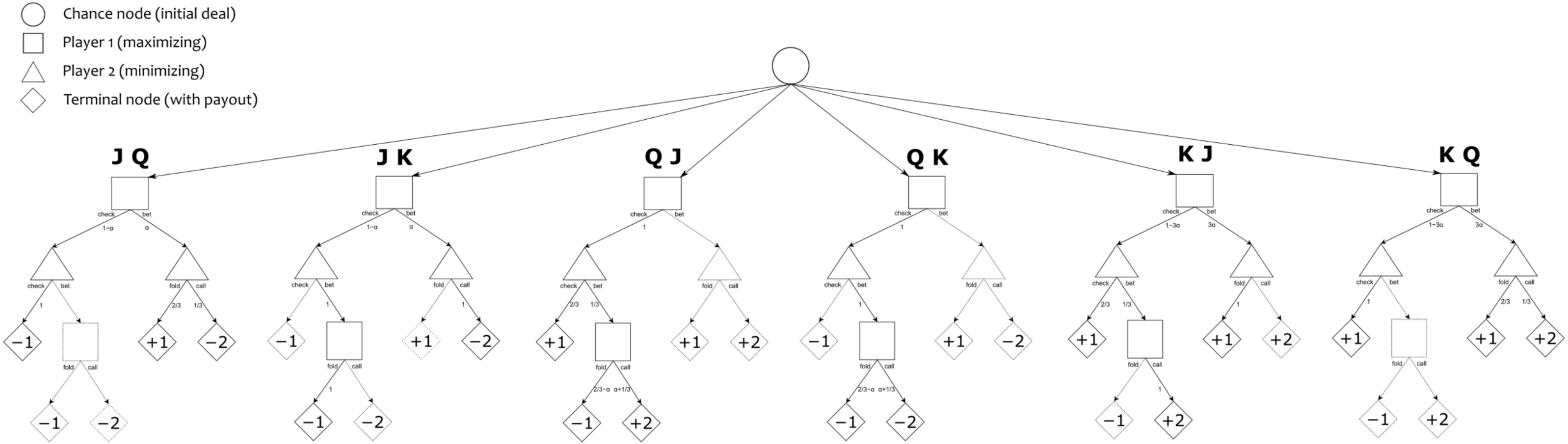
Extensive to Matrix Form



	2:*cc*	2:*cf*	2:*ff*	2:*fc*
1:*rr*	0	-1/6	1	7/6
1:*kr*	-1/3	-1/6	5/6	2/3
1:*rk*	1/3	0	1/6	1/2
1:*kk*	0	0	0	0

Exponential in number of info states!

A more interesting example: Kuhn Poker

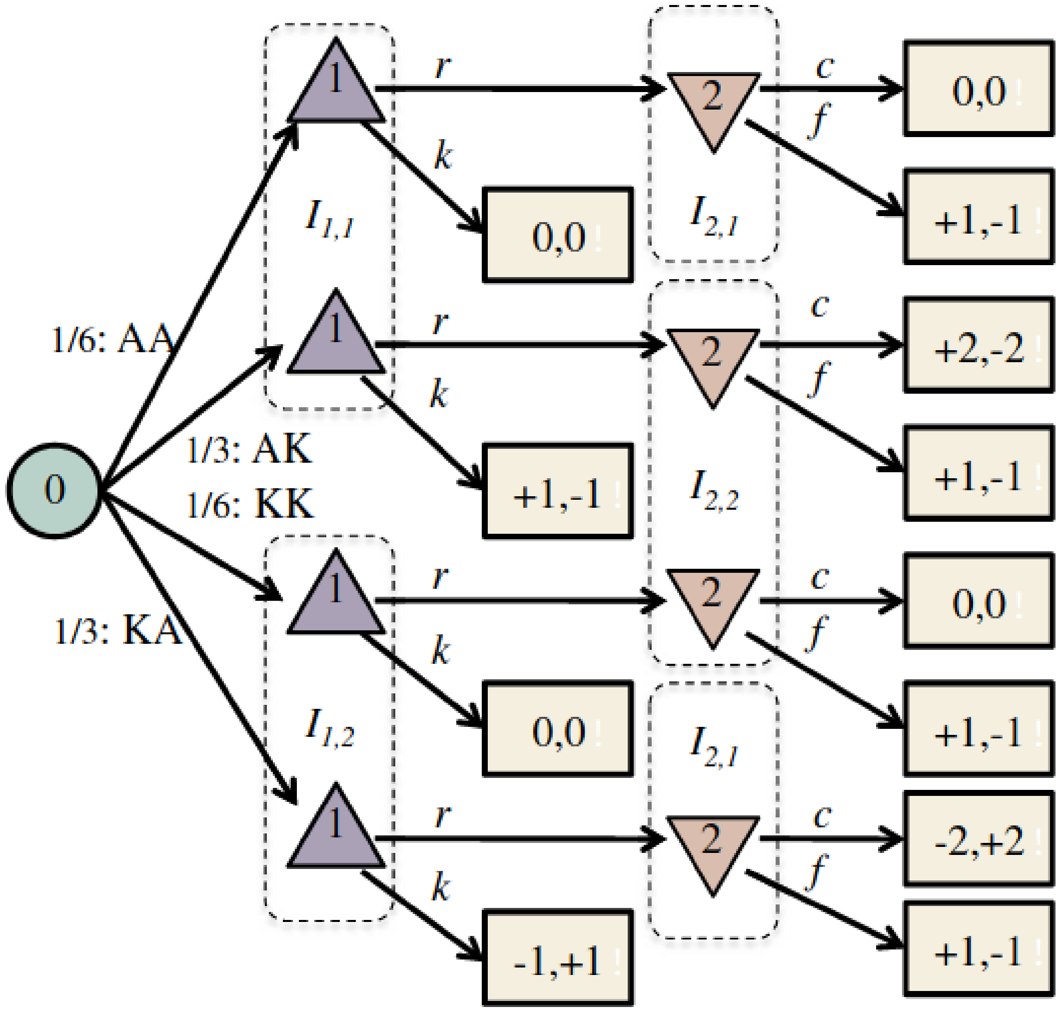


Fictitious Play in Extensive Form Games

Algorithm 2 General Fictitious Self-Play

function FICTITIOUSSELFPLAY(Γ, n, m)
  Initialize completely mixed π_1
  $\beta_2 \leftarrow \pi_1$
  $j \leftarrow 2$
  while within computational budget **do**
    $\eta_j \leftarrow$ MIXINGPARAMETER(j)
    $\mathcal{D} \leftarrow$ GENERATEDATA$(\pi_{j-1}, \beta_j, n, m, \eta_j)$
    for each player $i \in \mathcal{N}$ **do**
      $\mathcal{M}^i_{RL} \leftarrow$ UPDATERLMEMORY$(\mathcal{M}^i_{RL}, \mathcal{D}^i)$
      $\mathcal{M}^i_{SL} \leftarrow$ UPDATESLMEMORY$(\mathcal{M}^i_{SL}, \mathcal{D}^i)$
      $\beta^i_{j+1} \leftarrow$ REINFORCEMENTLEARNING$(\mathcal{M}^i_{RL})$
      $\pi^i_j \leftarrow$ SUPERVISEDLEARNING$(\mathcal{M}^i_{SL})$
    end for
    $j \leftarrow j + 1$
  end while
  return π_{j-1}
end function

function GENERATEDATA(π, β, n, m, η)
  $\sigma \leftarrow (1-\eta)\pi + \eta\beta$
  $\mathcal{D} \leftarrow n$ episodes $\{t_k\}_{1 \le k \le n}$, sampled from strategy profile σ
  for each player $i \in \mathcal{N}$ **do**
    $\mathcal{D}^i \leftarrow m$ episodes $\{t^i_k\}_{1 \le k \le m}$, sampled from strategy profile (β^i, σ^{-i})
    $\mathcal{D}^i \leftarrow \mathcal{D}^i \cup \mathcal{D}$
  end for
  return $\{\mathcal{D}^k\}_{1 \le k \le N}$
end function

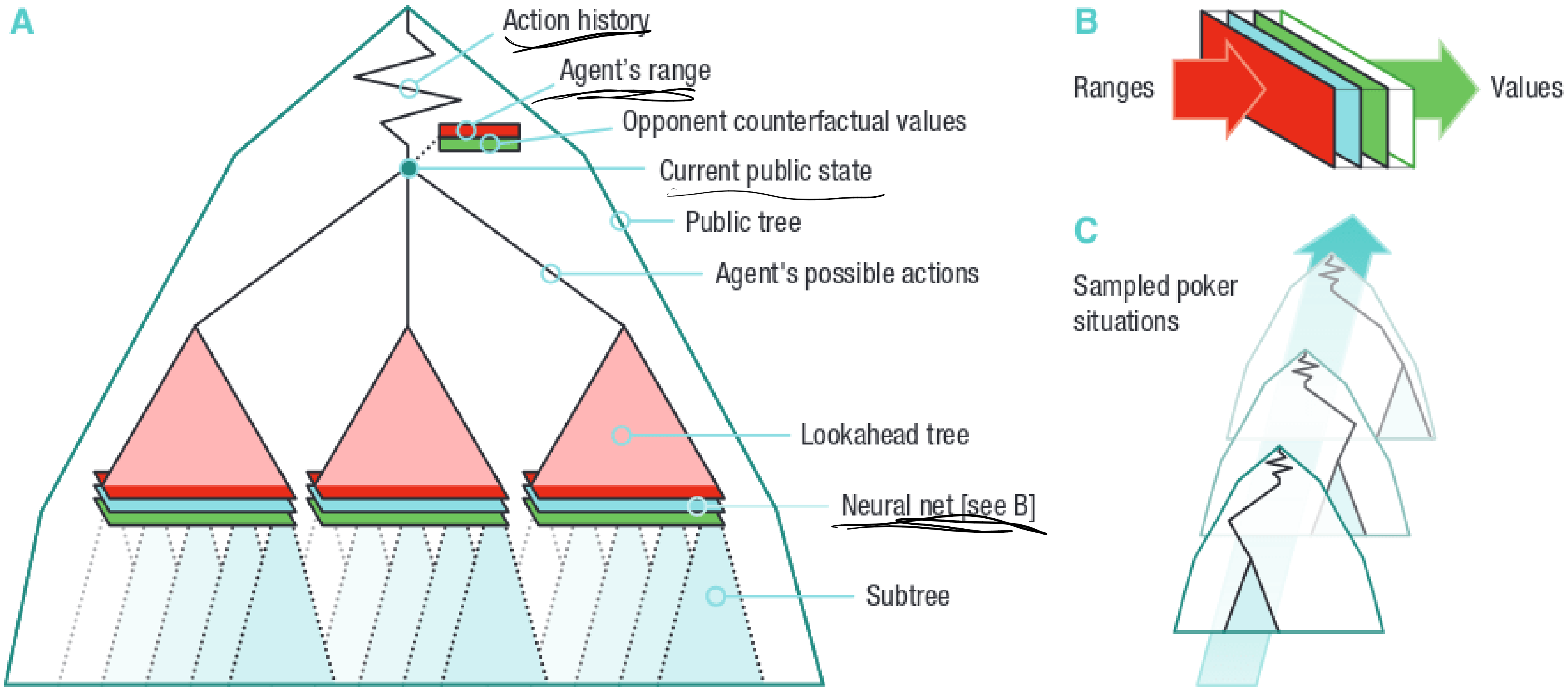


Heinrich et al. 2015 "Fictitious Self Play in Extensive-Form Games"

This slide not covered on exam

Deep Stack: Scaling to Heads Up No Limit Texas Hold 'Em

Counterfactual Regret Minimization + Deep Learning



Can game learning methods like CFR be used in Large POMGs?