Week 10 T9

Tuesday, 27 September 2022 2:54 PM

Offline Planning

- Policy learnt before runtime
- Used when state space is small
- Value iteration, Q-learning etc

Online planning

- Policy for current state learnt during runtime
- Used for large state space problems
- MCTS

General Online planning algorithm.

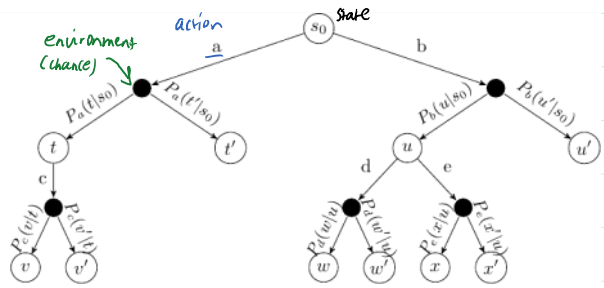
For each available action in current state:

1) Approximate Q-value through multiple simulations
2) Choose action with highest Q-value

Requires simulator or model to approximate/get transition probabilities

Monte Carlo Tree Search (MCTS)

Expectimax tree



Like a search tree, the expectimax tree is built incrementally

- Terminate search if complete tree is expanded, or some set time/node limit is reached
- Return action from current state with highest value.

4 steps to MCTS

While limit is not reached, repeat:

Select

Expand

Simulation

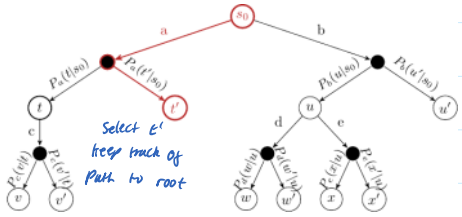
Backpropagate

Return best action

Selection

Select node that has not been fully expanded

(at least one child not yet explored)



While S is fully expanded:

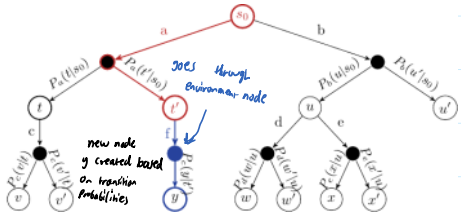
1) Choose action to perform (UCT look at later)

2) Use simulator / transition probabilities to determine new state

3) S ← new state

Expansion

Unless terminal / goal state, expand node by choosing an action and creating new child nodes for every possible new state



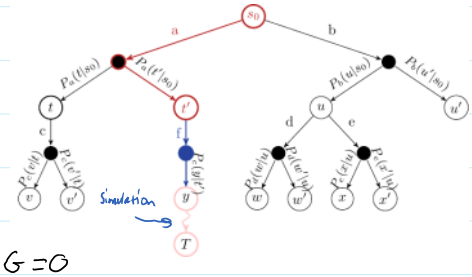
Choose action a to apply

Use simulator / transition probabilities to get new state s'

and reward r

Simulation

Perform random actions in simulator until reaching a terminal state



G = 0

t = 0

While S is not terminal:

1) Apply random action a

2) Use simulation / transition probabilities to get new state s' and reward r

3) G ← G + γ^t * r

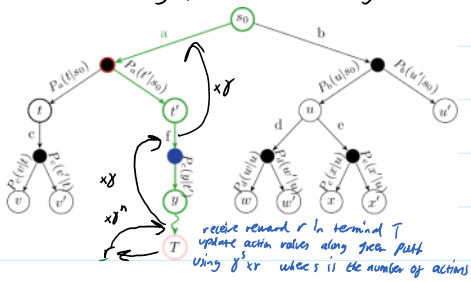
4) t ← t+1

5) S ← s'

After simulation, discard all simulated info except G and the first action taken

Back propagation

Observe reward r at terminal state. Update value V(S) for each node along path to root using discounted reward



Update Rule: $Q(S,a) = Q(S,a) + \frac{1}{N(S,a)} \times [r + \gamma G - Q(S,a)]$

While S exists: (repeat until after we apply to root node)

$N(S,a) \leftarrow N(S,a) + 1$ Increase # Times we see action (S,a)

$G \leftarrow r + \gamma G$ Add current reward and discount future reward

$Q(S,a) \leftarrow Q(S,a) + \frac{1}{N(S,a)}[G - Q(S,a)]$ Update Q value

$S \leftarrow$ Parent state ← Go up the tree 1 step

$a \leftarrow$ Parent action ↙

Upper Confidence Trees (UCT)

Strategy for selecting actions during selection step

Choose action a maximising:

$$Q(S,a) + 2C_p\sqrt{\frac{2\ln N(S)}{N(S,a)}}$$

$C_p > 0$ determines weight of exploration

$N(S)$: # times visiting state S

$N(S,a)$: # times performing action (S,a)

Higher value of C_p encourages more exploration