Search

DFS
BFS
UCS
A*
IDS

\[ f(x) = g(x) + h(x) \]

**Fringe:**
- priority queue of states
- index in queue is \( f \) ten

prob w/ search in general:
- exponential expansion

prob w/ \( A^* \):
- finding a good heuristic
- whole fringe has to stay in memory
Adversarial Search

minimax algorithm
alpha-beta pruning

expectimax - probabilistic weighting

final state has utility - win or lose

evaluation fn - estimated value of current state

\[ \text{sum of weighted products of feature fn's.} \]

early computable

world is fully observable
Markov Decision Processes (MDPs)

Solved with Value Iteration

Q-Learning

World:

states $s \in S$
actions $a \in A$
transition $T(s,a,s')$
reward $R(s,a,s')$
$R(s)$
start state
terminal state(s)
$q$-state $(s,a)$
Utility:
Utility is a sequence of rewards

\[ U([r_0, r_1, r_2, \ldots]) = r_0 + r_1 + r_2 + \ldots \]
\[ 0 \leq r \leq 1 \]

Policy:
\( \pi \) action for each state
\( \pi^* \) maximizes utility
Fixed policy \( \rightarrow \) reflex agent
Bellman Eqn's

\[ V^*(s) = \max_a Q^*(s, a) \]

\[ Q^*(s, a) = \sum_{s'} T(s, a, s') \cdot \left[ R(s, a, s') + \gamma V^*(s') \right] \]

Value Iteration

\[ V_{i+1}(s) \leftarrow \max_a \sum_{s'} T(s, a, s') \cdot \left[ R(s, a, s') + \gamma V_i(s') \right] \]
Why Q-learning

- don't know P/T for s or R
- play game (choose actions)
- learn optimal policy
- learner makes choices along the way
- learner can trade off
  - take chances - exploring vs. exploiting
  - 1 - ε exploit

Q-learning alg.

- learning $Q^*(s,a)$
  - receive sample $(s, a, s', r)$
  - estimated $Q(s,a)$; update it.

$$Q^*(s,a) = \sum_{s'} T(s,a,s') \left[ R(s,a,s') + \max_{a'} Q^*(s',a') \right]$$

$V_i(s)$ value assuming current policy
A

6

4/10 \rightarrow 10
6/10 \rightarrow 0

6

\text{lucky}
"A" -> 6

"B" -> 5

0.5

"B" -> 0.5

lucky B

4/5 -> 10
1/5 -> 0

unlucky B

0.5

4/5 -> 10
1/5 -> 0

0 + 0.15 if lucky
4/5 -> unlucky
Choose B and get 0.

$\begin{align*}
K & = 0.32 \\
(1,1) & \to (0,2) \\
0.8 	imes 0.8 + 0.2 	imes 0.2 & = 0.64 + 0.04 = 0.68
\end{align*}$

$A \perp F$

$A \perp F$

Diagram:

[A diagram showing a network with nodes labeled A, S, F, R, H, and V, with arrows indicating connections.]