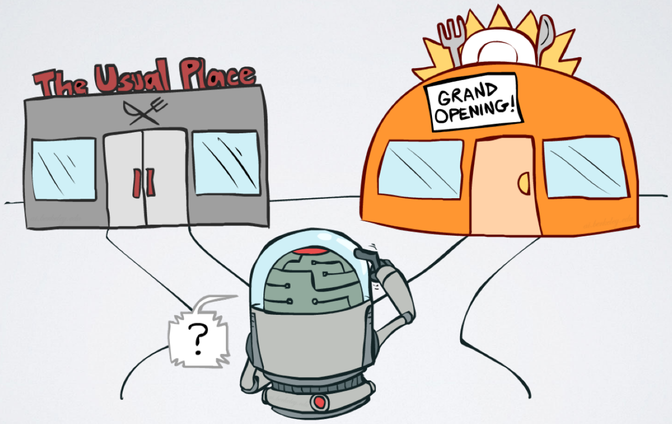
EXPLORATION VS. EXPLOITATION

CSE 511A: Introduction to Artificial Intelligence

Some content and images are from slides created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley.
All CS188 materials are available at http://ai.berkeley.edu.

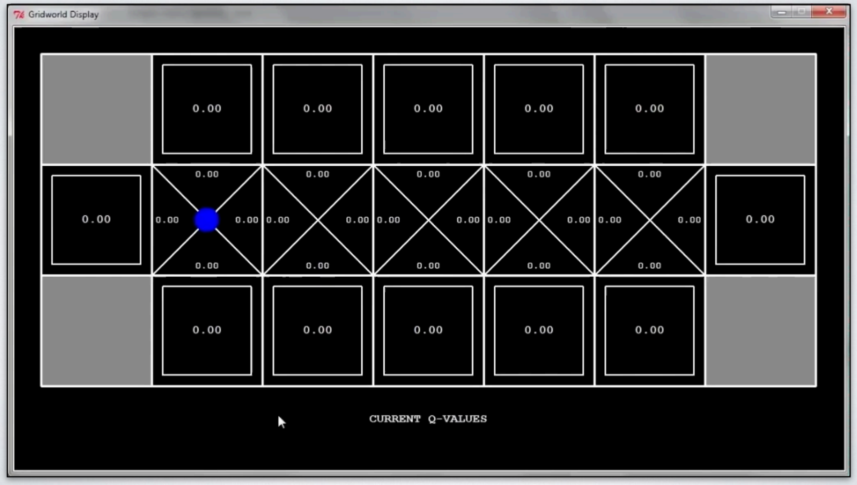
1

EXPLORATION VS. EXPLOITATION



2

EXPLORATION VS. EXPLOITATION

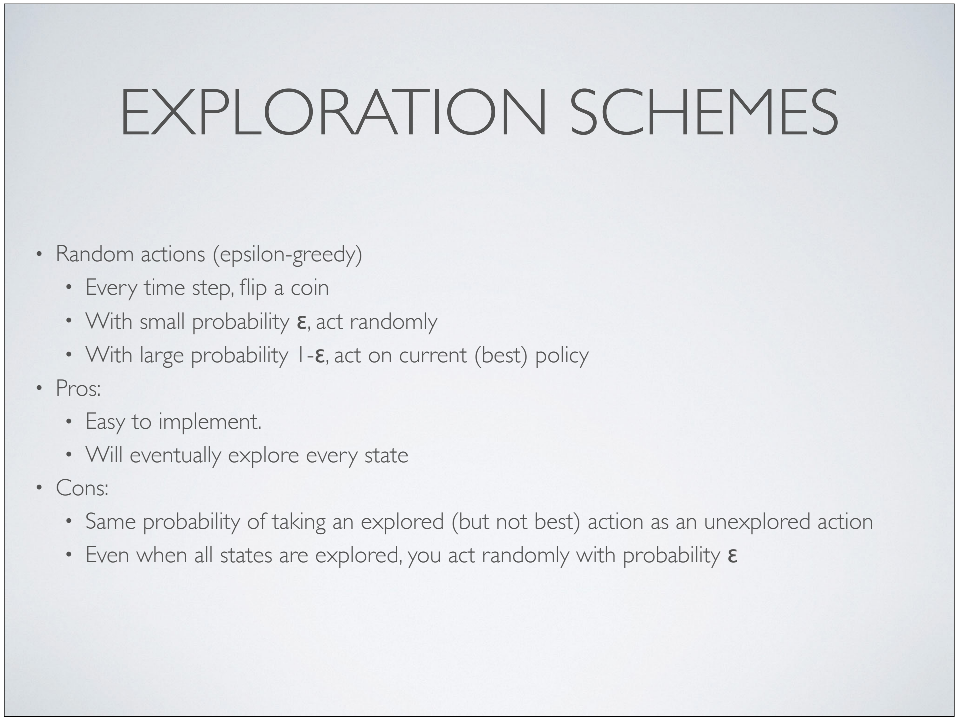


3

EXPLORATION SCHEMES

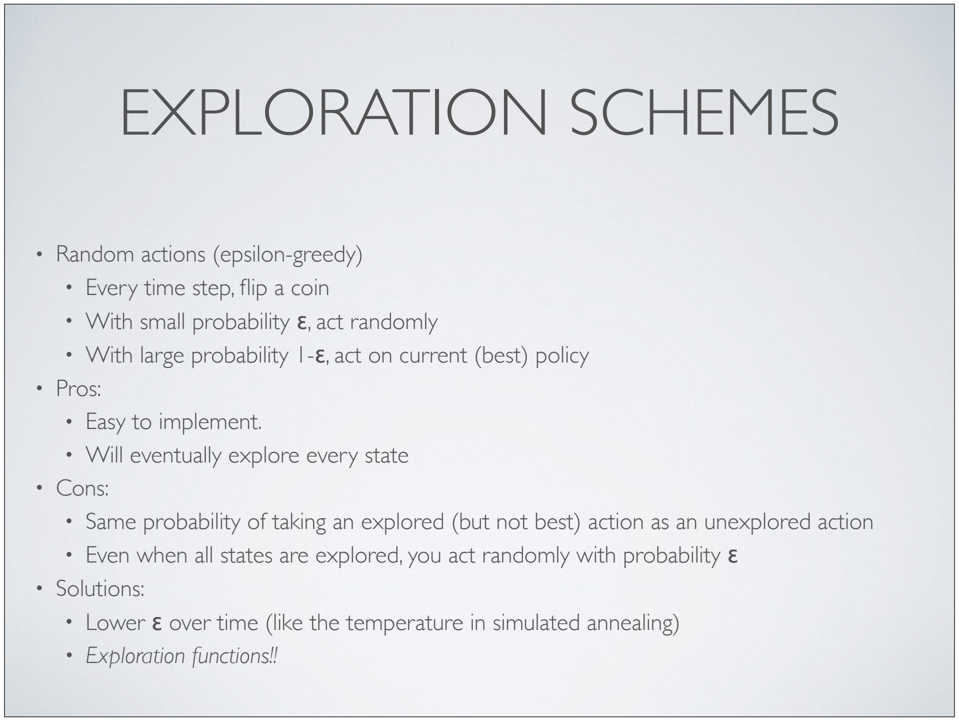
- Random actions (epsilon-greedy)
  - Every time step, flip a coin
  - With small probability **ε**, act randomly
  - With large probability 1-**ε**, act on current (best) policy

4



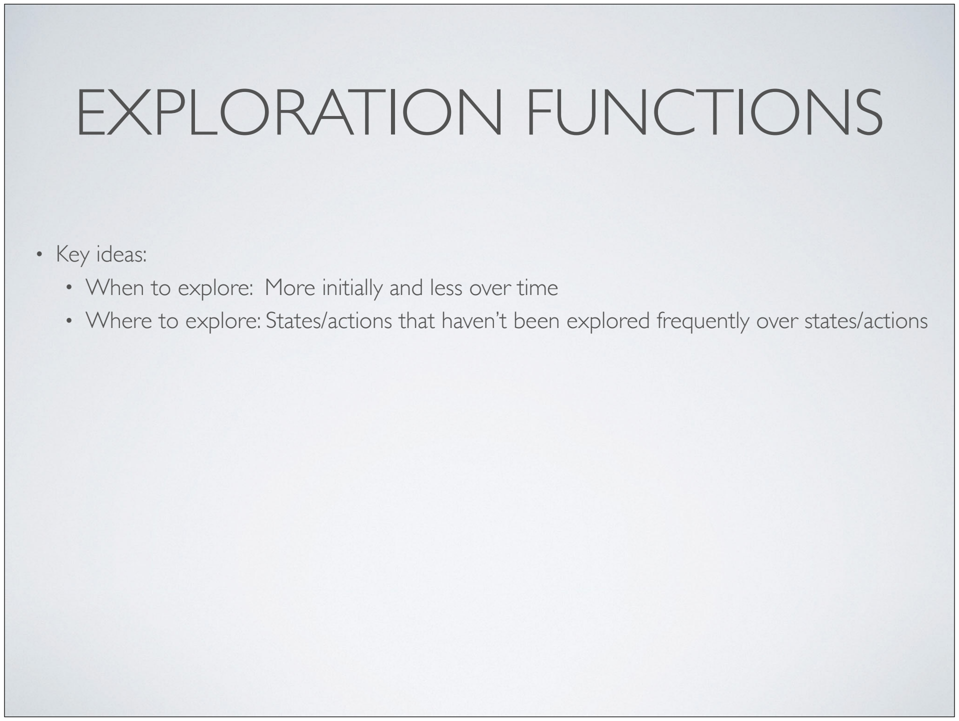
EXPLORATION SCHEMES

- Random actions (epsilon-greedy)
  - Every time step, flip a coin
  - With small probability **ε**, act randomly
  - With large probability 1-**ε**, act on current (best) policy
- Pros:
  - Easy to implement.
  - Will eventually explore every state
- Cons:
  - Same probability of taking an explored (but not best) action as an unexplored action
  - Even when all states are explored, you act randomly with probability **ε**



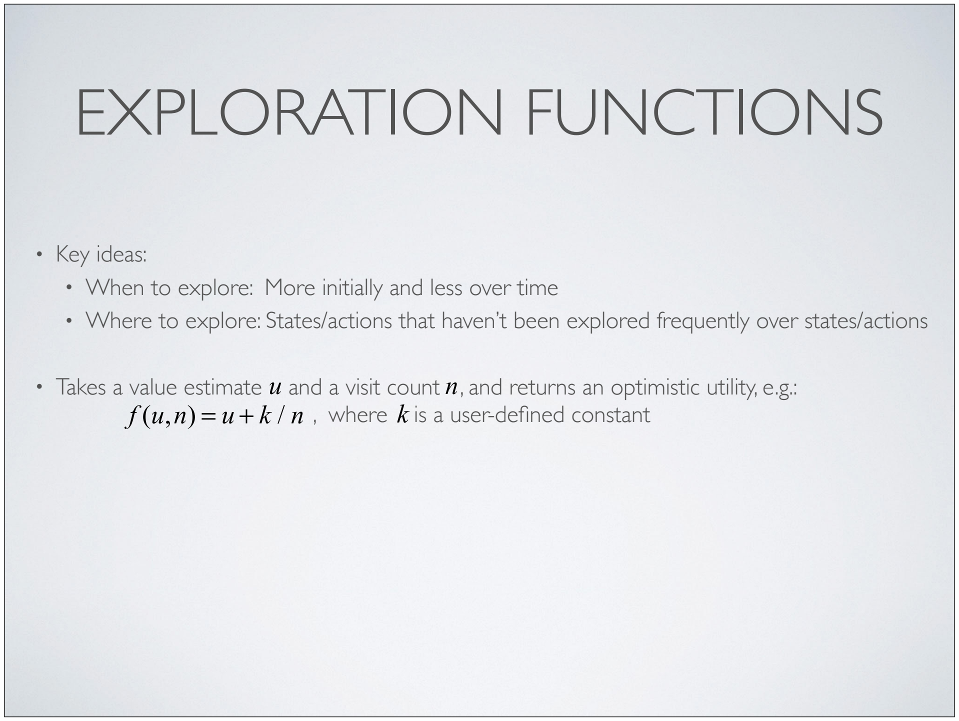
EXPLORATION SCHEMES

- Random actions (epsilon-greedy)
  - Every time step, flip a coin
  - With small probability **ε**, act randomly
  - With large probability 1-**ε**, act on current (best) policy
- Pros:
  - Easy to implement.
  - Will eventually explore every state
- Cons:
  - Same probability of taking an explored (but not best) action as an unexplored action
  - Even when all states are explored, you act randomly with probability **ε**
- Solutions:
  - Lower **ε** over time (like the temperature in simulated annealing)
  - *Exploration functions!!*



EXPLORATION FUNCTIONS

- Key ideas:
  - When to explore: More initially and less over time
  - Where to explore: States/actions that haven't been explored frequently over states/actions



EXPLORATION FUNCTIONS

- Key ideas:
  - When to explore: More initially and less over time
  - Where to explore: States/actions that haven't been explored frequently over states/actions

- Takes a value estimate u and a visit count n, and returns an optimistic utility, e.g.:
  $f(u,n) = u + k/n$, where k is a user-defined constant

EXPLORATION FUNCTIONS

- Key ideas:
  - When to explore: More initially and less over time
  - Where to explore: States/actions that haven't been explored frequently over states/actions

- Takes a value estimate u and a visit count n, and returns an optimistic utility, e.g.:
  $f(u,n) = u + k / n$, where k is a user-defined constant
- Regular Q-value update:
  $$Q(s,a) = (1-\alpha) \cdot Q(s,a) + \alpha \cdot \left[r + \gamma \max_{a'} Q(s',a') \right]$$
- Modified Q-value update:
  $$Q(s,a) = (1-\alpha) \cdot Q(s,a) + \alpha \cdot \left[r + \gamma \max_{a'} f(Q(s',a'), N(s',a')) \right]$$

  where $N(s',a')$ indicates the number of times action a' has been taken from state s'

EXPLORATION FUNCTIONS

- Key ideas:
  - When to explore: More initially and less over time
  - Where to explore: States/actions that haven't been explored frequently over states/actions

- Takes a value estimate u and a visit count n, and returns an optimistic utility, e.g.:
  $f(u,n) = u + k / n$, where k is a user-defined constant
- Regular Q-value update:
  $$Q(s,a) = (1-\alpha) \cdot Q(s,a) + \alpha \cdot \left[r + \gamma \max_{a'} Q(s',a') \right]$$
- Modified Q-value update:
  $$Q(s,a) = (1-\alpha) \cdot Q(s,a) + \alpha \cdot \left[r + \gamma \max_{a'} f(Q(s',a'), N(s',a')) \right]$$

  where $N(s',a')$ indicates the number of times action a' has been taken from state s'

- Important observation: $\lim_{n \to \infty} f(u,n) = u$