

# COMP90054 – Week 11 tutorial

Prepared by Zijie Xu. Last updated: 18 October 2023

## Policy iteration

### Idea

- A policy-based method that learns a policy directly (c.f. value-based methods)
- Perform policy updates by iterating on the policy

### Steps

- Policy evaluation: Given a policy  $\pi$ , for all  $s \in S$ , evaluate  $\pi(s)$  to find an action chosen by the policy in state  $s$  in the current iteration

- Solve a system of  $|S|$  equations for  $V^\pi(s) = Q^\pi(s, \pi(s))$

- For each  $s \in S$ :

$$V^\pi(s) = \sum_{s' \in S} P_{\pi(s)}(s' | s) [r(s, a, s') + \gamma V^\pi(s')]$$

- For terminal states,  $V^\pi(s) = 0$

- Unlike in Bellman equations, treat all  $V^\pi(s)$  as variables to solve, do not swap for values from previous iteration

- Policy update: Improve the policy  $\pi(s)$

- $\pi(s) = \operatorname{argmax}_{a \in A(s)} Q^\pi(s, \pi(s))$

- Repeat the evaluation and update steps until  $\pi(s)$  has no more updates

## Reward shaping

### Idea

- Give an additional reward to algorithm to help it converge more quickly
- The additional reward  $F(s, s')$  is given to agent whenever it transitions from  $s$  to  $s'$ 
  - $F$  provides heuristic domain knowledge to the problem that is typically manually programmed.
  - $F(s, s') > 0$  incentivises actions that transitions agent from  $s$  to  $s'$
- $r + F(s, s')$  is the shaped reward

### Potential-based reward shaping

- A particular type of reward shaping
- $F(s, s') = \gamma \Phi(s') - \Phi(s)$ 
  - $\Phi$  is the potential function
  - $\Phi(s)$  provides the potential of state  $s$
- e.g. Normalised Manhattan distance as a potential function in GridWorld
  - $\Phi(s) = 1 - \frac{|x_g - x_s| + |y_g - y_s|}{width + height - 2}$
  - $x_g, y_g$ : coordinates of goal cell
  - $x_s, y_s$ : coordinates of the agent in state  $s$
  - $-2$  to account for zero indexing of coordinates in normalisation