Finding the Shortest Path Using Reinforcement Learning

LINGFENG ZHANG, WENHAN HAO, & TIANMING QIU

Group: applied-rl17 / E3

1 Consice Introduction of Project

This project focuses on finding a shortest path in a map with many cross roads. The agent has to make an optimal decision to turn into a correct or best direction at each road crossing, namely the state in terms of Markov Decision Process.

2 Markov Decision Process Description

Obviously, the states in this model are the different road crossings. State transition, namely the transition of different road crossing, is adopted as a stochastic process. The modelling of this issue satisfies the Markov Decision Process (MDP). First, it has the Markov property. And the transition from road crossing C_t to next crossing C_{t+1} is independent of all the previous states and actions. No matter which road crossing the Epuck was in or which actions it takes before the road crossing C_t, the probability that it transits from current state to next state would be never influenced. In another words, it can be represented by: $P(s_{t+1}) = P(s_t, a_t)$. The current state captures all that is relevant about the world in order to predict what the next state will be. Secondly, it is clear to see that the state transition is related to action. Epuck can choose to turn right, turn left, go ahead or backward, and the consequence will be dependent on the chosen actions at current state.

A Markov decision process is a discrete-time state transition system. It can be described formally with 5 components as a 5-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$ as below:

$\mathcal{S}$**: States** First, it has a finite set of states. These states will play the role of outcomes in the decision theoretic approach, as well as providing whatever information is necessary for choosing actions.

$\mathcal{A}$**: Action** A small finite set includes turn left, turn right, go forward and backward.

$\mathcal{S}$**: Transition probability** The transition probabilities describe the dynamics of the world. They play the role of the next-state function in a problem-solving search, except that every state is thought to be a possible consequence of taking an action in a state. So, we specify, for each state s_t and action a_t, the probability that the next state will be s_{t+1}. Now we just model the transition probability as the simplest one: All of them equal to one. That means if and only if Epuck takes a specific action (e.g. turn right) at the current action, it will transit into a definitely deterministic state.

$\mathcal{R}$**: Reward** In this model we will assign a reward '+10' at the destination state and '0' at the other states. Also at the 'blind alley' it will get a very poor reward such as '-100'.

γ γ is a discout factor here we select as 0.9.

3 Algorithm

to be written to be written

4 Milestones

to be written to be written

References