Pickup and Delivery Reactive Agent: Implementation of MDP

Christophe Marciot & Titouan Renard

September 30, 2020

1 Problem definition

1.1 Markov Decision Process

In a MDP current state is know with certainty, but the reward of transition is not. A MDP is defined by :

$$\textit{A reward function:} \quad \overbrace{R(s,a) \to \mathbb{R}}^{\textit{Where } s \textit{ denotes a state and } a \textit{ an action}}$$

$$\textit{A probabilistic state transition table:} \quad \overbrace{T(s,a,s') = p(s'|s,a)}^{\textit{Where } s' \textit{ denotes the state the action leads to}}$$

The goal of the process is to find a policy π such that *the average reward is maximized.*

1.2 The Pickup and Delivery Problem

Agents exist in a static environment (a model of Switzerland's road network) described by a graph. Nodes of the graph are called *cities* and it's (weighted) edges are called *roads.*

The pickup and delivery problem is described by a series of tasks spread over the topology, the transportation tasks are described by:

1. Pickup city
2. Delivery city
3. Reward in CHF

1.2.1 Existing tables

The dataset usable for learning is described by two probability tables :

1. $P_{table}(i,j)$: the probability of a task for city j to be present in city i
2. $R_{table}(i,j)$: the average reward given when a task is transported from city i to city j

2 Solving MDP

We denote *the value of a state* s as $V(s)$. This value represents "*the potential rewards from this state onwards*". In order to ensure $V(s_i) < \infty \; \forall i$ (and make the problem solvable) we introduce a *discount factor* $\gamma \in [0...1[$.

$$V(s_i) = R(s_i) + \gamma \cdot V(T(s_i), a(s_i))$$