RL Silver 1
3. The Reinforcement Learning Problem

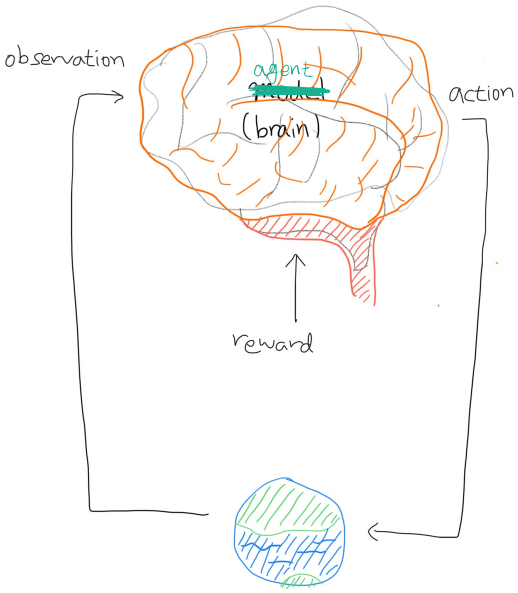
Rewards

scalar feedback
how well agent doing

no intermedial goal?

manoeuvres : 전술
trajectory : 궤도

reward examples
헬리콥터 곡예 비행
+ : 정해진 궤도를 따랐을 때
- : 망가졌을 때
Backgammon : 백개먼 보드게임
+- : 승패
투자 포트폴리오
+- : 은행에서 받은 돈
발전소 관리
휴머노이드 걷기
아타리 게임들



Sequential Decision Making
"selcet actions to maximise total future reward"
not greedy algorithm

History : the sequence of observation + action + reward (by time)

state : summary of the information for next action

$$S_t = f(H_t)$$

Environment State S_t^e ?
: the environment's private representation
-> we don't see that

Agent State S_t^a
: the agent's internal representation
we can build any function
-> actually used in RL

Markov state (= information state)
S(1~t)로 찾은 S(t+1)과 St로 찾은 S(t+1)이 같다고 가정
1~t의 history를 통해 St를 찾고 St를 이용해서 t+1~의 history를 찾는 것과 동일
즉 St만 필요하다

헬리콥터 문제에서의 Markov는?
현재 위치, 속도, 각도 등
-> 10분 전의 위치, 속도, 각도는 전혀 중요하지 않음

history Ht == Markov

Full Observable Environments

we can see everything (best case)
agent state = environment state = information state
-> Markov decision process (MDP)

Partially Observable Environment

cannot see whole environment
ex1) robot camera vision cannot told its absolute location
ex2) poker agent only observes public cards

agent state != environment state

POMDP(partially observable MDP)

1. record whole data X
2. Beliefs of environment state from probabilty pre-states
3. RNN(Recurrent) new state from last state

components
1. Policy : how agent pick action - agent's behaviour
2. Value function : how good it is, how well done - prediction of future reward for select next action
3. Model : how agent think the environment works