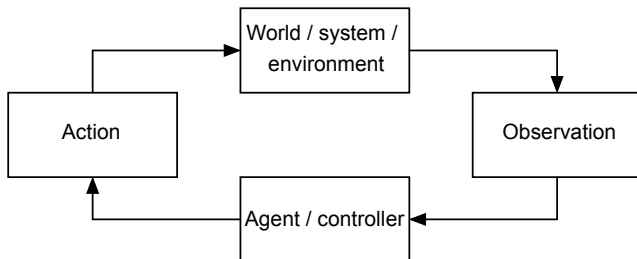


A short course on reinforcement learning

Hugo Touchette

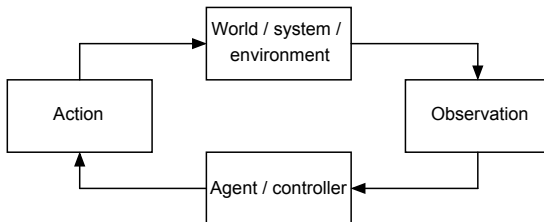
Department of Mathematical Sciences
Stellenbosch University, South Africa

Introduction



- Learn from interactions (exploration, trial and error)
- Reinforce good actions (exploitation)
- Goal-directed learning (reward)
- Find optimal way to act (optimal policy)
- Actions can affect future (delayed reward)
- Actions depend on situations (associativity)
- Uncertainty in environment and agent (probabilistic model)

Reward hypothesis



Practical version

All goals can be described by the maximisation of reward.

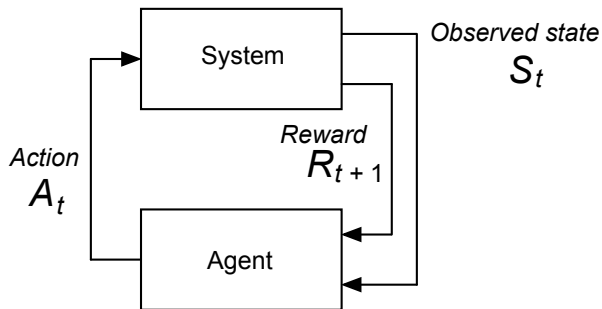
- Reward communicates what we want to achieve.
- Not how we want to achieve it (no instructions).

Strong version

[Silver et al. 2021]

Maximising reward is enough to drive behaviour that exhibits most if not all abilities studied in natural and artificial intelligence.

Simplified model



- Dynamics: $P(S_{t+1}, R_{t+1} | S_t, A_t)$
- Control: $P(A_t | S_t)$
- History/trajectory:

$S_0, A_0; R_1, S_1, A_1; R_2, S_2, A_2; \dots$

Comparison

- Supervised learning:
 - Training set of examples
 - Instructive feedback: indicate correct 'action to take' (input, label)
 - External supervisor
 - Extrapolate, generalize
 - Not interactive
- Reinforcement learning (3rd paradigm):
 - No examples of desired behaviour
 - No representative set of examples
 - Learn from experience, not training set
 - Evaluate actions to be taken (need for exploration)
 - Online, interactive learning
- Unsupervised learning:
 - Find hidden structure (e.g. classification boundary)
 - Not necessarily based on reward

Plan

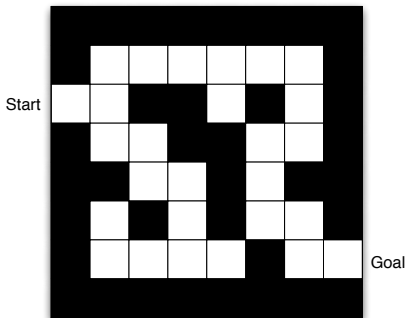
- Week 1: Markov processes
- Week 2: Markov decision processes
- Week 3: Dynamic programming and Bellman equations
- Week 4: Temporal difference algorithms: Sarsa and Q-learning

- Courseworks:
 - Applications (weeks 1-3)
 - 2D navigation with gym (week 4)

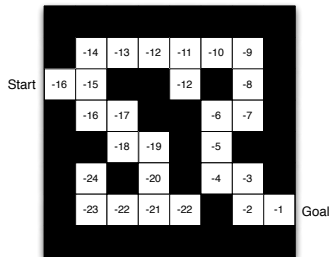
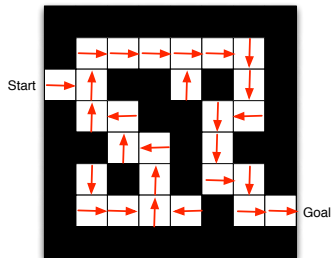
- Assistant: Umesh Kumar
`umesh.kumar@icts.res.in`
- Github: <https://github.com/HugoTouchette/ICTS-RL>
- See information sheet on github

Maze

[Source: David Silver's course]




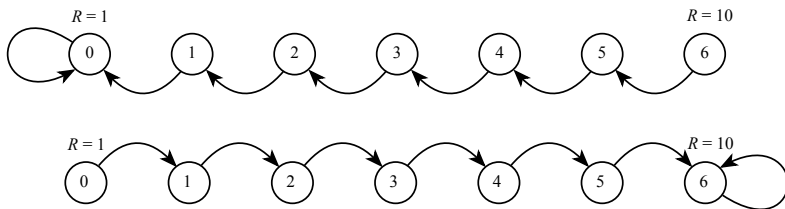
- State: Location
- Action: N, S, E, W
- Reward: -1 per step



Linear model

[Source: Mars rover example, Stanford RL course]

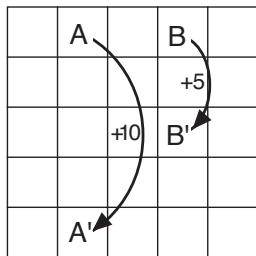
s_1	s_2	s_3	s_4	s_5	s_6	s_7
						



- State: Location
- Actions: Move left or right
- Rewards: +1 from state s_1 (0), +10 from state s_7 (6)

Gridworld model

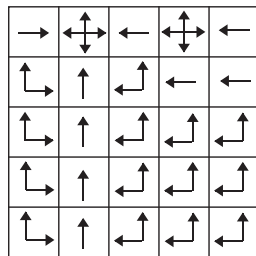
[Source: Sutton and Barto, Example 3.6]



Gridworld

22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*

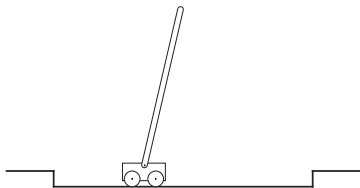


π_*

- State: Location
- Actions: N, S, E, W
- Rewards: -1 if leaving grid, $+10$ from A to A' , $+5$ from B to B'

Cartpole

[Source: Sutton and Barto, Example 3.5]



- State: Stick angle θ_t
- Actions: Impulse right or left
- Dynamics: Physics 101
- Rewards:
 - 0 if $|\theta| < \eta$ (balanced)
 - -1 otherwise (unbalanced)
- Optimal policy: ?