Machine Learning for Networking

Reinforcement Learning

Session 10 – Q-learning II

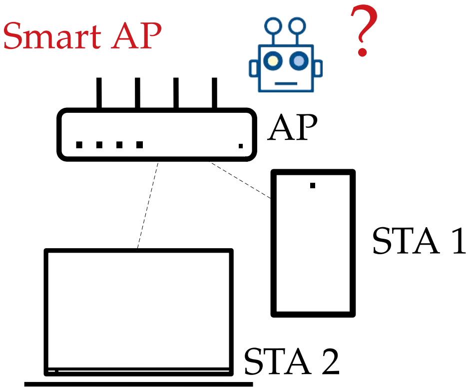
Boris Bellalta: boris.bellalta@upf.edu

Contents

- Q-learning: step by step, 2nd part
- Q-learning exercises
- Exercise: video congestion control → What is the effect of rewards?

States & Actions

channels 1 2



States (4)

- s1: 1,E
- s2: 1,B
- s3: 2,E
- s4: 2,B

actions

- a_1=Stay in the same channel
- a_2=Switch to the other channel

Q-learning

- Table to store the pair (state,action), i.e. the Q-table

State	Stay	Switch
1,E (s1)		
1,B (s2)		
2,E (s3)		
2,B (s4)		

- A mechanism to explore the state space
  - Epsilon-greedy (for example)

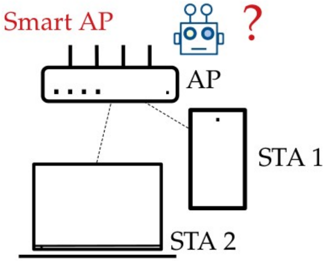
- A way to update the Q-table (from wikipedia) – Bellman's equation

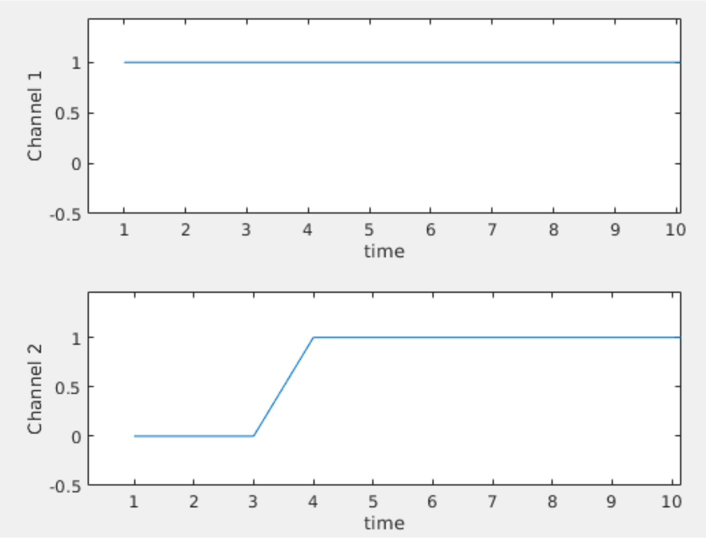
$$Q^{new}(s_t, a_t) \leftarrow \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \overbrace{\bigg(\underbrace{\underbrace{r_t}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_a Q(s_{t+1}, a)}_{\text{estimate of optimal future value}}}_{\text{new value (temporal difference target)}} - \underbrace{Q(s_t, a_t)}_{\text{old value}} \bigg)}^{\text{temporal difference}}$$

Case I

channels 1 2

α=0.2
γ=0.9



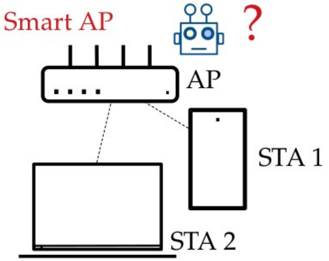


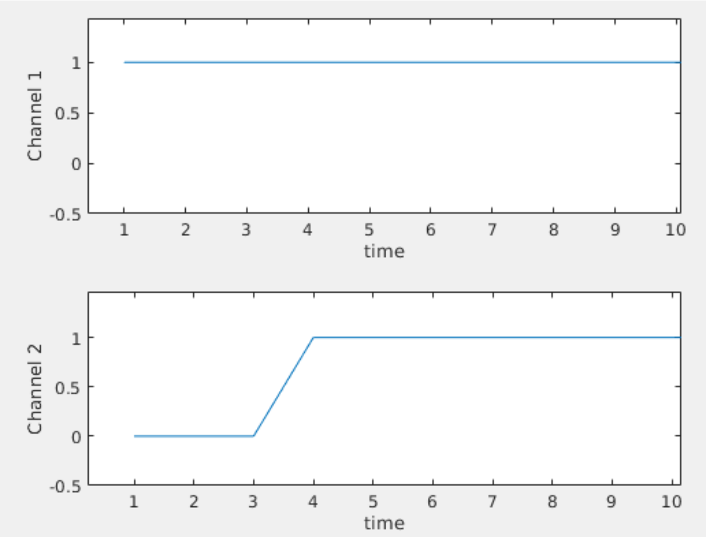

```
I have no info at the start
     0.0000e+000     0.0000e+000
     0.0000e+000     0.0000e+000
     0.0000e+000     0.0000e+000
     0.0000e+000     0.0000e+000
Iteration | Current State
     1.0000e+000     2.0000e+000
Explore? | Action (1 = stay | 2 = switch)
     0.0000e+000     1.0000e+000
Next state | Reward (1 = well done | -1 = :( )
     2.0000e+000    -1.0000e+000
(1-alfa) * Q_table(state(t),action(t)) + alfa * ( reward(t) + gamma * max(Q_table(state(t+1),:))) =
   800.0000e-003     0.0000e+000   200.0000e-003    -1.0000e+000   900.0000e-003     0.0000e+000
-----Q-table-----
     0.0000e+000     0.0000e+000
  -200.0000e-003     0.0000e+000
     0.0000e+000     0.0000e+000
     0.0000e+000     0.0000e+000
-----------------
```

Case I

channels 1 2

α=0.2
γ=0.9



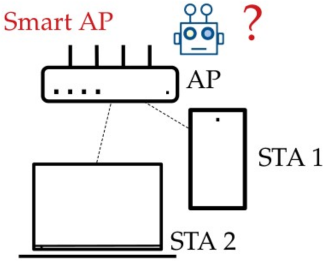


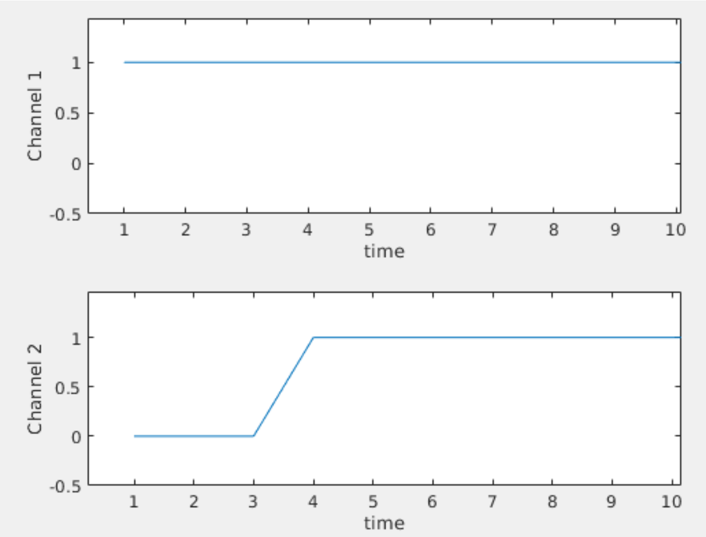

```
-----Q-table-----
     0.0000e+000      0.0000e+000
  -200.0000e-003      0.0000e+000
     0.0000e+000      0.0000e+000
     0.0000e+000      0.0000e+000
-----------------
Iteration | Current State
     2.0000e+000      2.0000e+000
Explore? | Action (1 = stay | 2 = switch)
     0.0000e+000      2.0000e+000
Next state | Reward (1 = well done | -1 = :( )
     3.0000e+000      1.0000e+000
(1-alfa) * Q_table(state(t),action(t)) + alfa * ( reward(t) + gamma * max(Q_table(state(t+1),:))) =
   800.0000e-003      0.0000e+000   200.0000e-003     1.0000e+000   900.0000e-003     0.0000e+000
-----Q-table-----
     0.0000e+000      0.0000e+000
  -200.0000e-003    200.0000e-003
     0.0000e+000      0.0000e+000
     0.0000e+000      0.0000e+000
-----------------
```

Case I

channels 1 2

α=0.2
γ=0.9



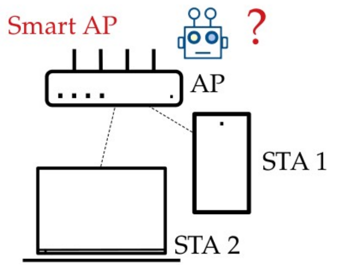


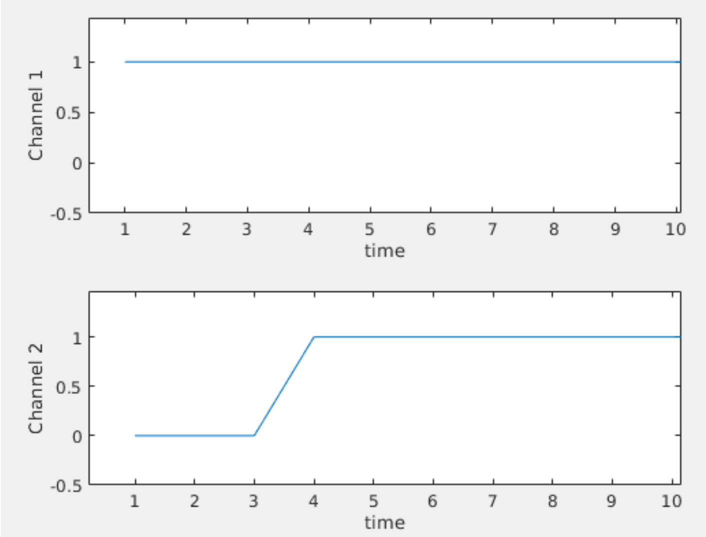

```
-----Q-table-----
     0.0000e+000      0.0000e+000
  -200.0000e-003    200.0000e-003
     0.0000e+000      0.0000e+000
     0.0000e+000      0.0000e+000
-----------------
Iteration | Current State
     3.0000e+000      3.0000e+000
Explore? | Action (1 = stay | 2 = switch)
     0.0000e+000      1.0000e+000
Next state | Reward (1 = well done | -1 = :( )
     4.0000e+000     -1.0000e+000
(1-alfa) * Q_table(state(t),action(t)) + alfa * ( reward(t) + gamma * max(Q_table(state(t+1),:))) =
   800.0000e-003      0.0000e+000    200.0000e-003     -1.0000e+000    900.0000e-003      0.0000e+000
-----Q-table-----
     0.0000e+000      0.0000e+000
  -200.0000e-003    200.0000e-003
  -200.0000e-003      0.0000e+000
     0.0000e+000      0.0000e+000
-----------------
```

Case I

channels 1 2

α=0.2
γ=0.9



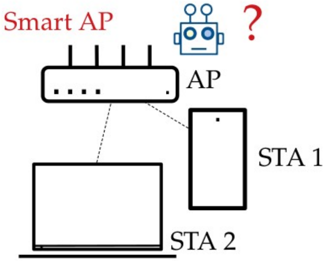


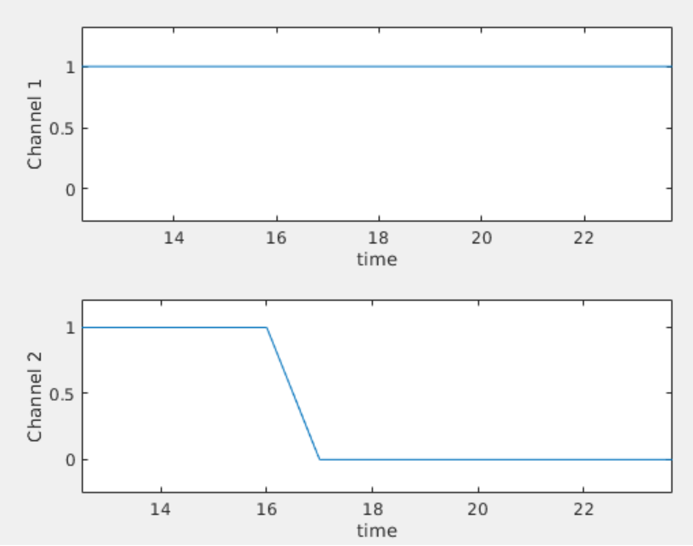

```
-----Q-table-----
     0.0000e+000      0.0000e+000
  -200.0000e-003    200.0000e-003
  -200.0000e-003      0.0000e+000
     0.0000e+000      0.0000e+000
-----------------
Iteration | Current State
     4.0000e+000      4.0000e+000
Explore? | Action (1 = stay | 2 = switch)
     0.0000e+000      1.0000e+000
Next state | Reward (1 = well done | -1 = :( )
     4.0000e+000     -1.0000e+000
(1-alfa) * Q_table(state(t),action(t)) + alfa * ( reward(t) + gamma * max(Q_table(state(t+1),:))) =
   800.0000e-003      0.0000e+000    200.0000e-003     -1.0000e+000    900.0000e-003      0.0000e+000
-----Q-table-----
     0.0000e+000      0.0000e+000
  -200.0000e-003    200.0000e-003
  -200.0000e-003      0.0000e+000
  -200.0000e-003      0.0000e+000
-----------------
Iteration | Current State
     5.0000e+000      4.0000e+000
Explore? | Action (1 = stay | 2 = switch)
     0.0000e+000      2.0000e+000
Next state | Reward (1 = well done | -1 = :( )
     2.0000e+000     -1.0000e+000
(1-alfa) * Q_table(state(t),action(t)) + alfa * ( reward(t) + gamma * max(Q_table(state(t+1),:))) =
   800.0000e-003      0.0000e+000    200.0000e-003     -1.0000e+000    900.0000e-003    200.0000e-003
-----Q-table-----
     0.0000e+000      0.0000e+000
  -200.0000e-003    200.0000e-003
  -200.0000e-003      0.0000e+000
  -200.0000e-003   -164.0000e-003
-----------------
```

Case I

channels 1 2

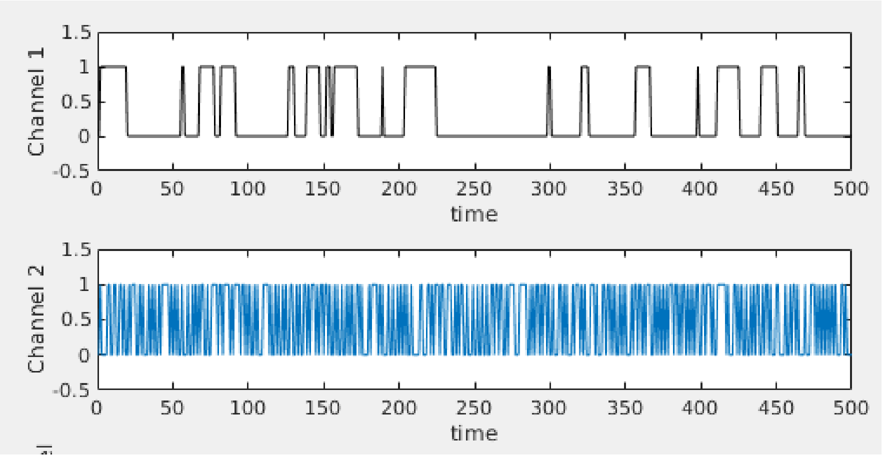
α=0.2
γ=0.9

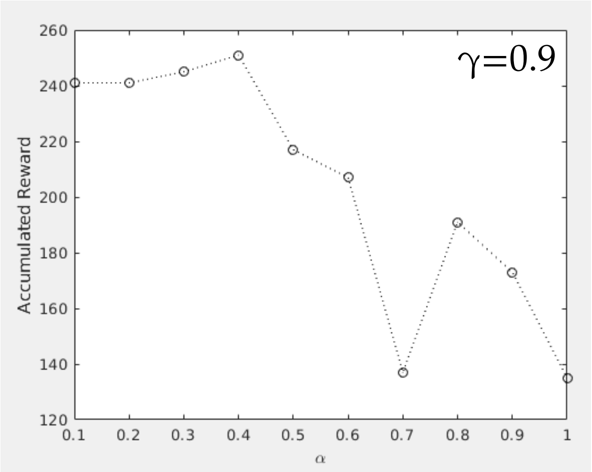


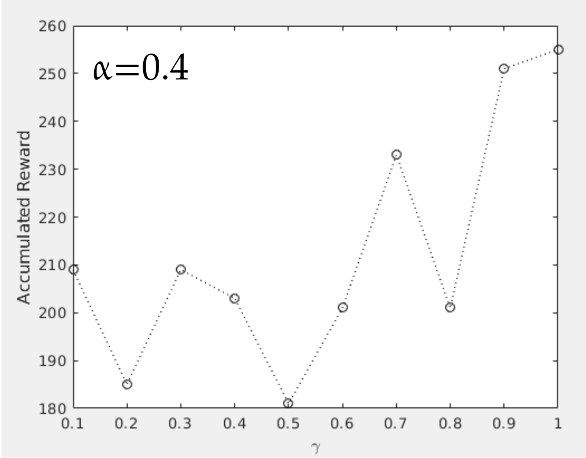



```
-----Q-table-----
      0.0000e+000      0.0000e+000
  -709.9346e-003  -219.2809e-003
  -200.0000e-003  -239.4706e-003
  -581.7296e-003  -680.0567e-003
-----------------
Iteration | Current State
     19.0000e+000      2.0000e+000
Explore? | Action (1 = stay | 2 = switch)
      0.0000e+000      2.0000e+000
Next state | Reward (1 = well done | -1 = :( )
      3.0000e+000      1.0000e+000
(1-alfa) * Q_table(state(t),action(t)) + alfa * ( reward(t) + gamma * max(Q_table(state(t+1),:))) =
   800.0000e-003  -219.2809e-003   200.0000e-003      1.0000e+000   900.0000e-003  -200.0000e-003
-----Q-table-----
      0.0000e+000      0.0000e+000
  -709.9346e-003   -11.4247e-003
  -200.0000e-003  -239.4706e-003
  -581.7296e-003  -680.0567e-003
-----------------
Iteration | Current State
     20.0000e+000      3.0000e+000
Explore? | Action (1 = stay | 2 = switch)
      0.0000e+000      1.0000e+000
Next state | Reward (1 = well done | -1 = :( )
      3.0000e+000      1.0000e+000
(1-alfa) * Q_table(state(t),action(t)) + alfa * ( reward(t) + gamma * max(Q_table(state(t+1),:))) =
   800.0000e-003  -200.0000e-003   200.0000e-003      1.0000e+000   900.0000e-003  -200.0000e-003
-----Q-table-----
      0.0000e+000      0.0000e+000
  -709.9346e-003   -11.4247e-003
     4.0000e-003  -239.4706e-003
  -581.7296e-003  -680.0567e-003
-----------------
Iteration | Current State
     21.0000e+000      3.0000e+000
Explore? | Action (1 = stay | 2 = switch)
      0.0000e+000      1.0000e+000
Next state | Reward (1 = well done | -1 = :( )
      3.0000e+000      1.0000e+000
(1-alfa) * Q_table(state(t),action(t)) + alfa * ( reward(t) + gamma * max(Q_table(state(t+1),:))) =
   800.0000e-003      4.0000e-003   200.0000e-003      1.0000e+000   900.0000e-003      4.0000e-003
-----Q-table-----
      0.0000e+000      0.0000e+000
  -709.9346e-003   -11.4247e-003
   203.9200e-003  -239.4706e-003
  -581.7296e-003  -680.0567e-003
-----------------
```

Case II - Test of α and γ

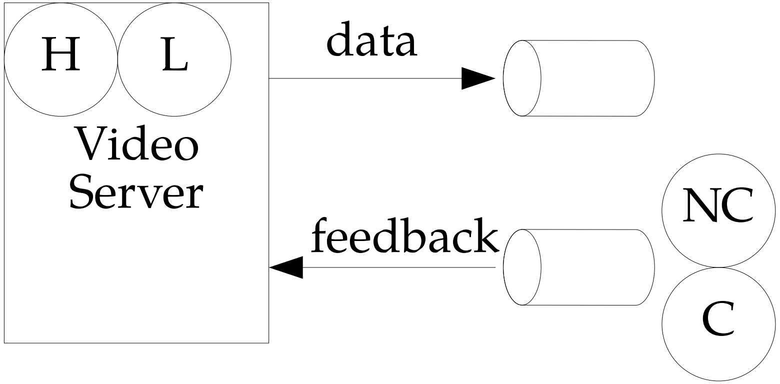






$$Q^{new}(s_t, a_t) \leftarrow \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \overbrace{\Bigg(\underbrace{\underbrace{r_t}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_a Q(s_{t+1}, a)}_{\text{estimate of optimal future value}}}_{\text{new value (temporal difference target)}} - \underbrace{Q(s_t, a_t)}_{\text{old value}} \Bigg)}^{\text{temporal difference}}$$

Exercise: Congestion Control Video Server



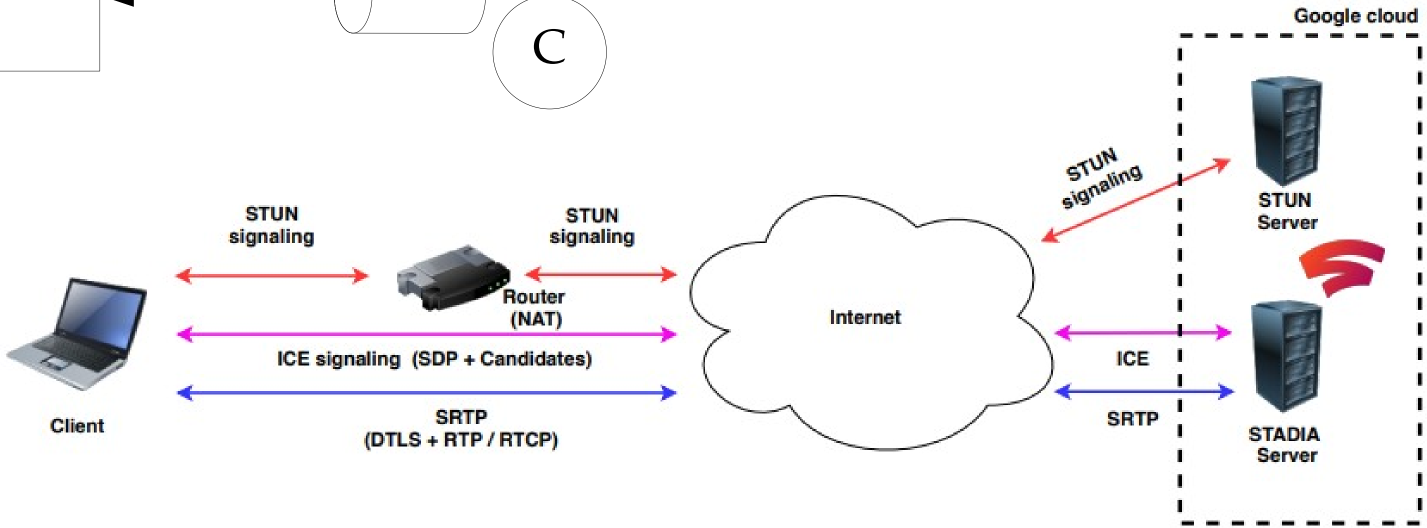
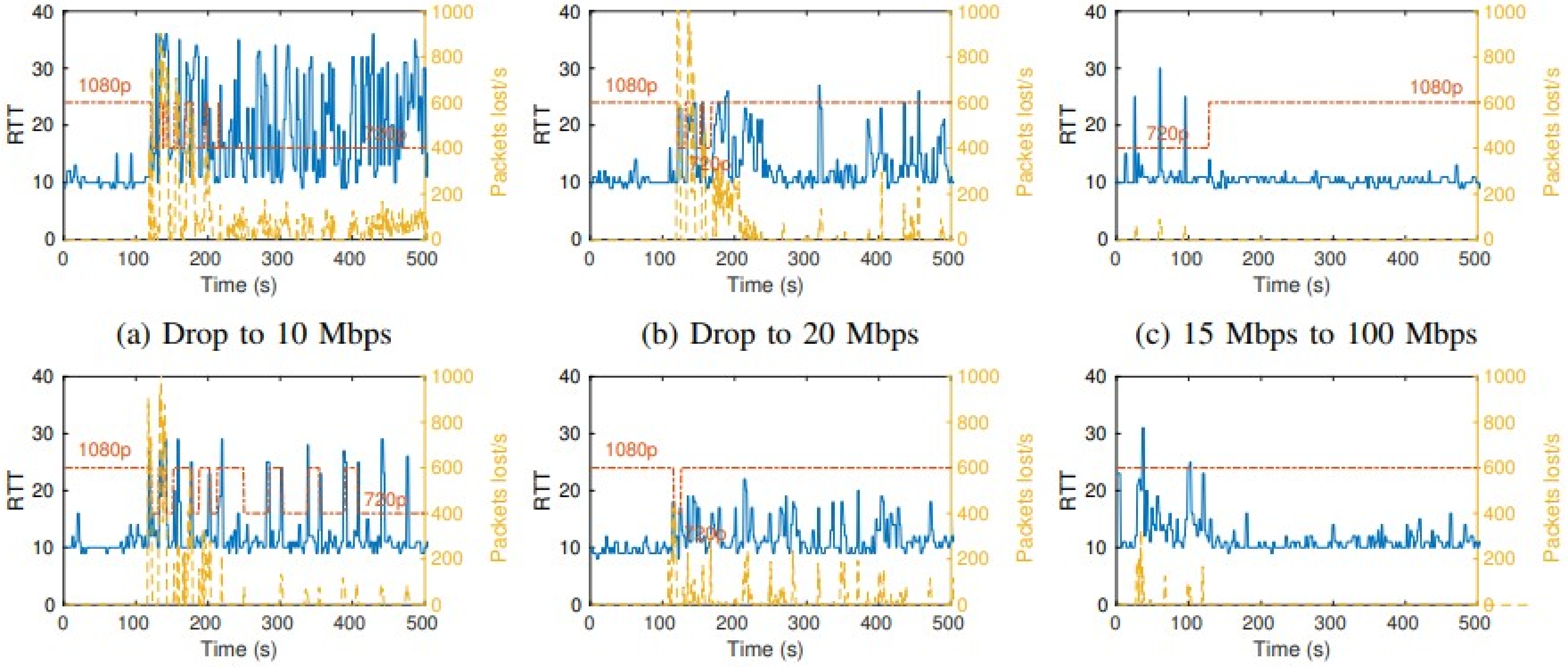


Fig. 1: Google's Stadia: Main components and data streams.

Google Stadia: WebRTC



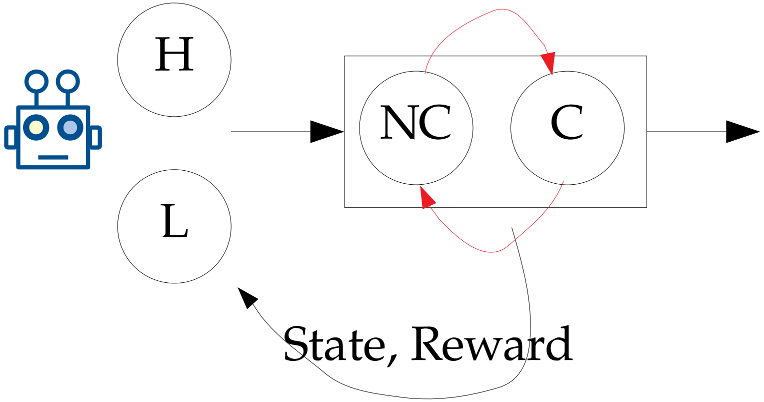
(a) Drop to 10 Mbps
(b) Drop to 20 Mbps
(c) 15 Mbps to 100 Mbps
(d) Drop to 15 Mbps
(e) Drop to 30 Mbps
(f) 20 to 100 Mbps

Fig. 11: Round Trip Time (continuous line), video packets lost (dashed line) and resolution (dash-dotted line).

Exercise: Congestion Control Video Server

- Implement the described scenario, try different channel transitions probabilities
  - Video sender: H (high quality), L (low quality), Channel: NC (Non congested), C (congested)
- Test assuming the agent makes random decisions at each iteration.
- Implement Q-learning, and evaluate if there is any gain. Fine tune Q-learning.

Channel (p(nc→c)=0.01;p(c→nc)=0.04)



Rewards

(H,NC) R1 = 1
(H,C) R2 = 0
(L,NC) = 0.75
(L,C) = 1

Activity

- Investigate what is the effect of the rewards (change it, trying to make them consistent ...)
- Will the agent learn a different strategy if

(H,NC) R1 = 1
(H,C) R2 = 0
(L,NC) = 0.25
(L,C) = 0.75