École Nationale Supérieure d'Électronique, d'Informatique,
d'Hydraulique et des Télécommunications

Final Project : Load balancing

Iliass Sijelmassi

INP ENSEEIHT
3ème année SN

Sommaire

1 MDP **1**

1.1 Policy Evaluation 1

1.2 Optimal control 2

2 Tabular Model-Free control **4**

2.1 Policy Evaluation 4

2.2 Optimal control 5

3 Model-free control with Value Function/Policy approximation **7**

Partie 1

MDP

1.1 Policy Evaluation

First we assume the random policy that dispatches every job with probability 0.5 to either queue 1 and 2.

$$V(Q1, Q2) = E[Q1 + Q2 + *V(Q1', Q2')] \tag{1.1}$$

$$V(Q1, Q2) = \gamma\,(0.5(\lambda + \mu1 \cdot V(Q1 - 1, Q2) + \mu2 \cdot V(Q1, Q2 - 1)) + 0.5(1 - \lambda - \mu1 - \mu2) \cdot V(Q1, Q2) + Q1 + Q2) \tag{1.2}$$

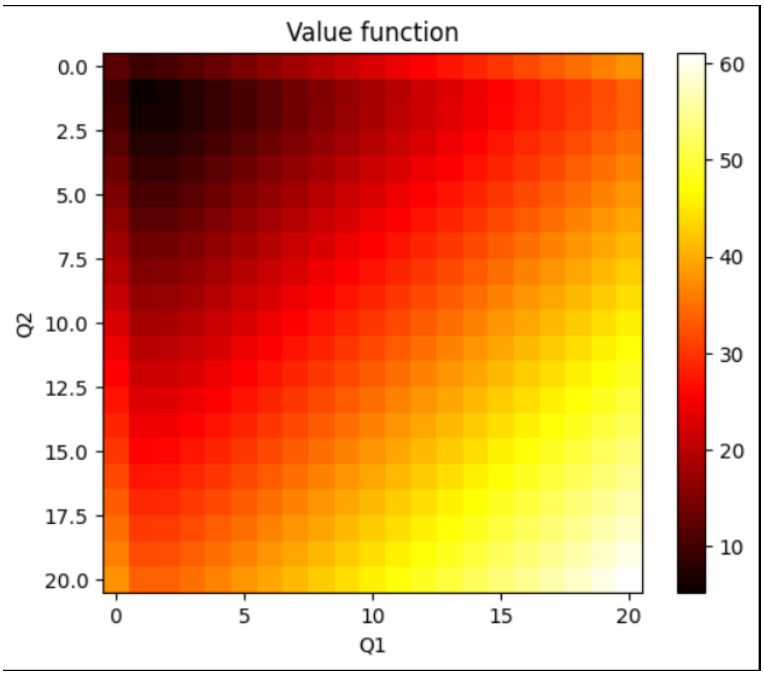

FIGURE 1.1

1.2 Optimal control

In this part we are asked to find the optimal policy to dispatch incoming jobs.
$V^*(Q1, Q2) = \min[V^*(Q1+1, Q2), V^*(Q1, Q2+1)] + \gamma$ * (Q1 + Q2 + 1), where
$V^*(Q1+1, Q2) = \lambda$ * $V^*(Q1+2, Q2) + (1 - \lambda$ - $\mu_1)$ * $V^*(Q1+1, Q2) + \mu_1$ * $V^*(Q1, Q2)$
$V^*(Q1, Q2+1) = \lambda$ * $V^*(Q1, Q2+2) + (1 - \lambda$ - $\mu_2)$ * $V^*(Q1, Q2+1) + \mu_2$* $V^*(Q1, Q2)$

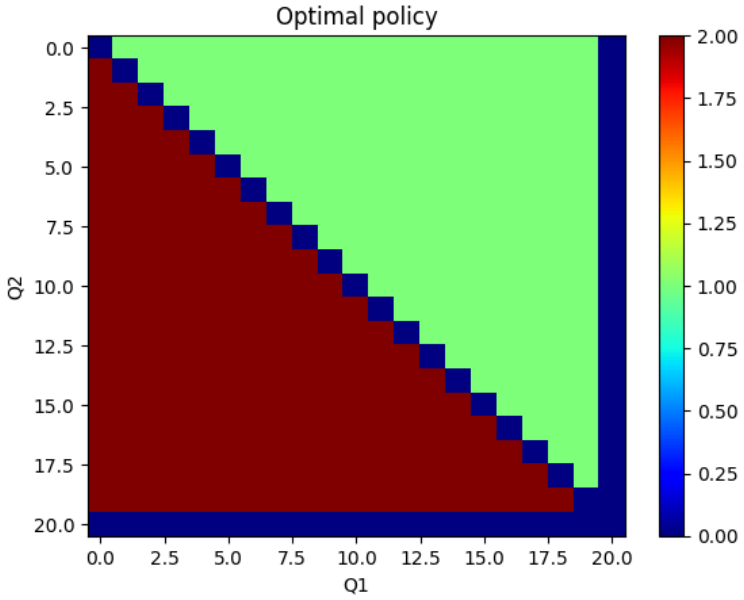

FIGURE 1.2

$V^*(Q_1, Q_2) = \min_{a_i \in 1,2} Q_1 + Q_2 + \gamma \sum_{Q_1', Q_2'} P(Q_1', Q_2' | Q_1, Q_2, a_i) V^*(Q_1', Q_2')$
(1.3)

where $V^*(Q_1, Q_2)$ is the optimal value function at state (Q_1, Q_2), $P(Q_1', Q_2' | Q_1, Q_2, a_i)$ is the transition probability of moving from state (Q_1, Q_2) to (Q_1', Q_2') given action a_i, and γ is the discount factor.

A one-step improvement of the random policy involves updating the policy to take the action that gives the maximum expected discounted return, based on the current estimated value function. The updated policy will only differ from the optimal policy by at most one action in one state.

After the one-step improvement, we would observe that the value function improves from the previous iteration and that the new policy is closer to the optimal policy. This improvement in the value function indicates that the updated policy is better than the previous random policy.

If we repeat this process, we would get a sequence of improving policies until we reach the optimal policy. This process of iteratively improving the policy is called Policy Iteration.

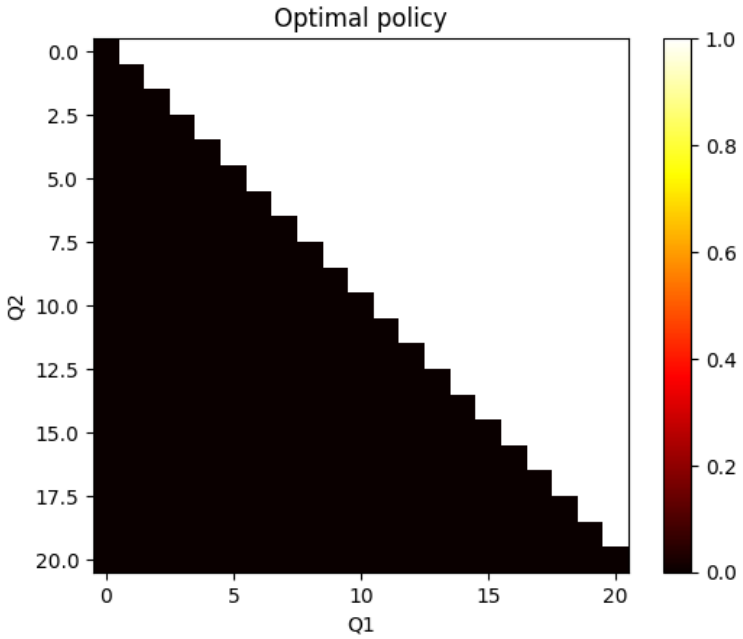

FIGURE 1.3

We observe the optimal Policy.

Chapitre 2

Tabular Model-Free control

2.1 Policy Evaluation

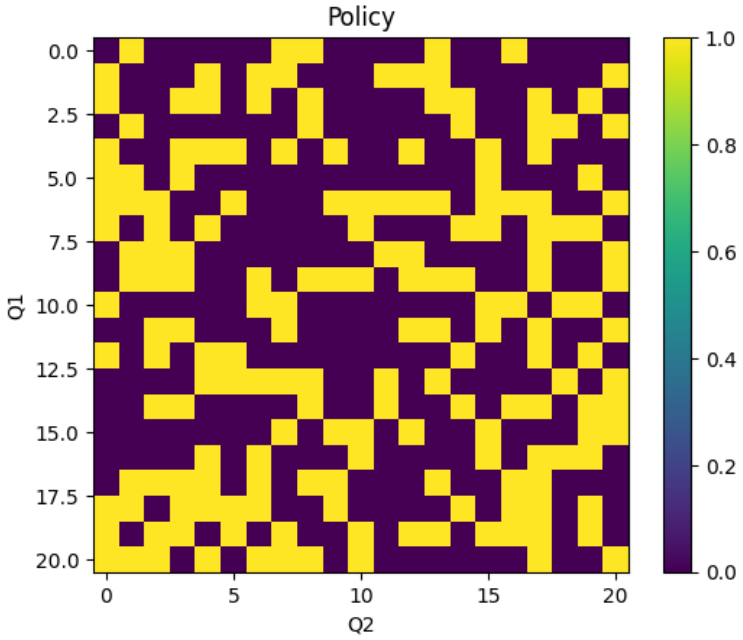

FIGURE 2.1

Completely different from section 1 : not as effective
Alternatives for α_n : put a low value constant : $\alpha = 0.1$ per example.

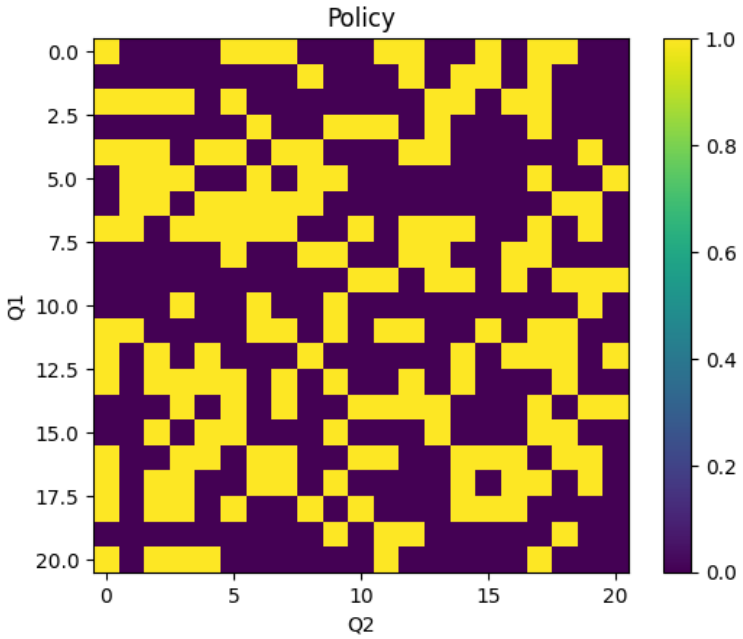

FIGURE 2.2

2.2 Optimal control

$Q_learninghasbeenimplemented(seecode).Itisstillnotaseffectiveasthecodeofsection1.$

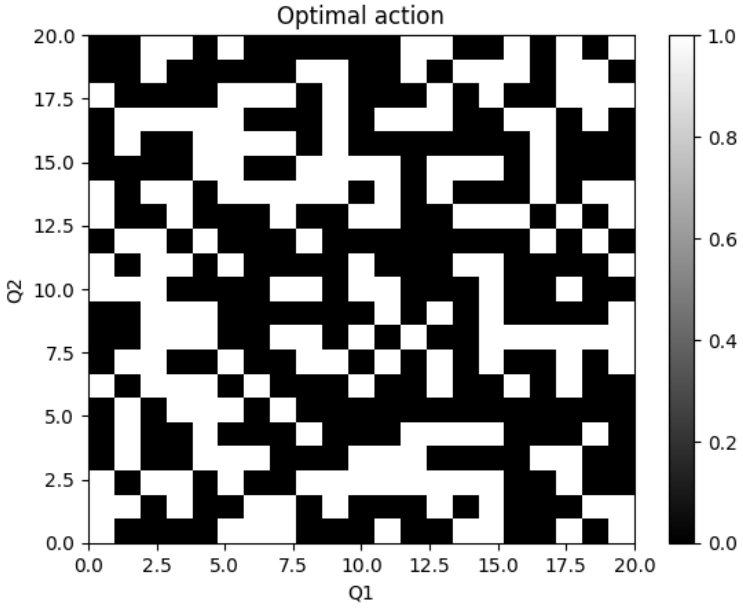

FIGURE 2.3 – optimal action as a function of the state (Q1, Q2) for $\alpha = 0.1$
We take here a constant of $\alpha_n = 1/(\mathrm{n}^\gamma)$

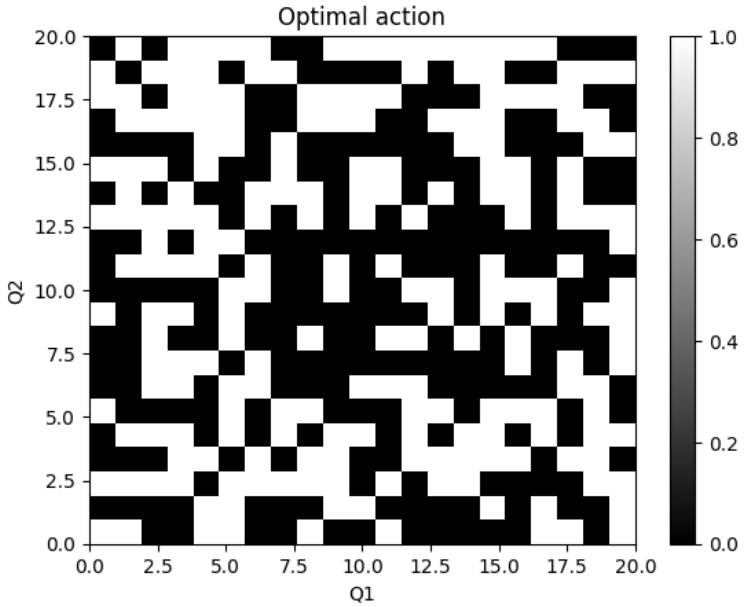

FIGURE 2.4 – optimal action as a function of the state (Q1, Q2) given $\alpha_n = 1/(\mathrm{n}^\gamma)$

We can see that it converges faster. However it is still not as effective as the policy of Section 1.

Chapitre 3

Model-free control with Value Function/Policy approximation