etis

UNIVERSITÉ
de Cergy-Pontoise

Reinforcement learning

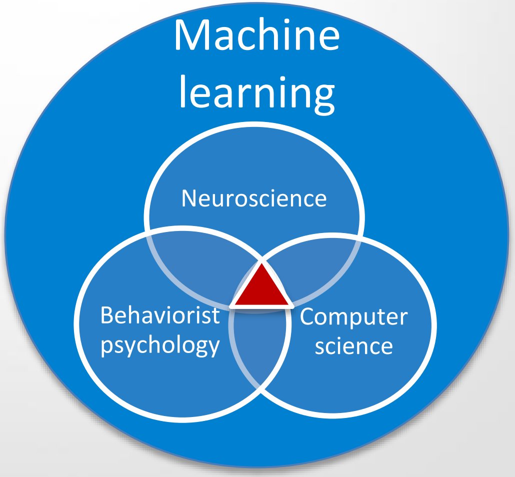
Bibliographic search by Matthieu Vilain

Tutor : Pierre Andry

May 2017

Réseau Figure
CURSUS MASTER EN INGÉNIERIE

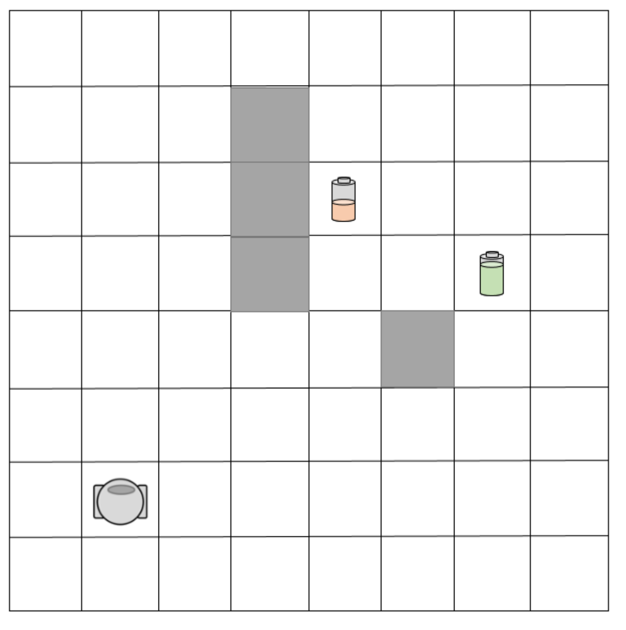
What is reinforcement learning ?



Way of programming agents by reward and punishment without needing to specify how the task is to be achieved

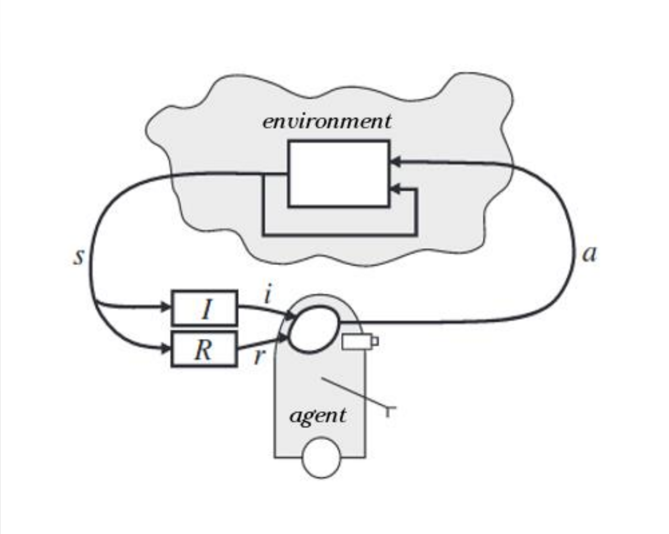
- TD Learning - R.Sutton - 1988
- Q Learning - C.Watkins - 1989-1992

Exemple



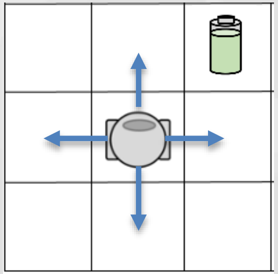
How does it work ?

The agent and his environment



The environment must be divided into several states

Exemple

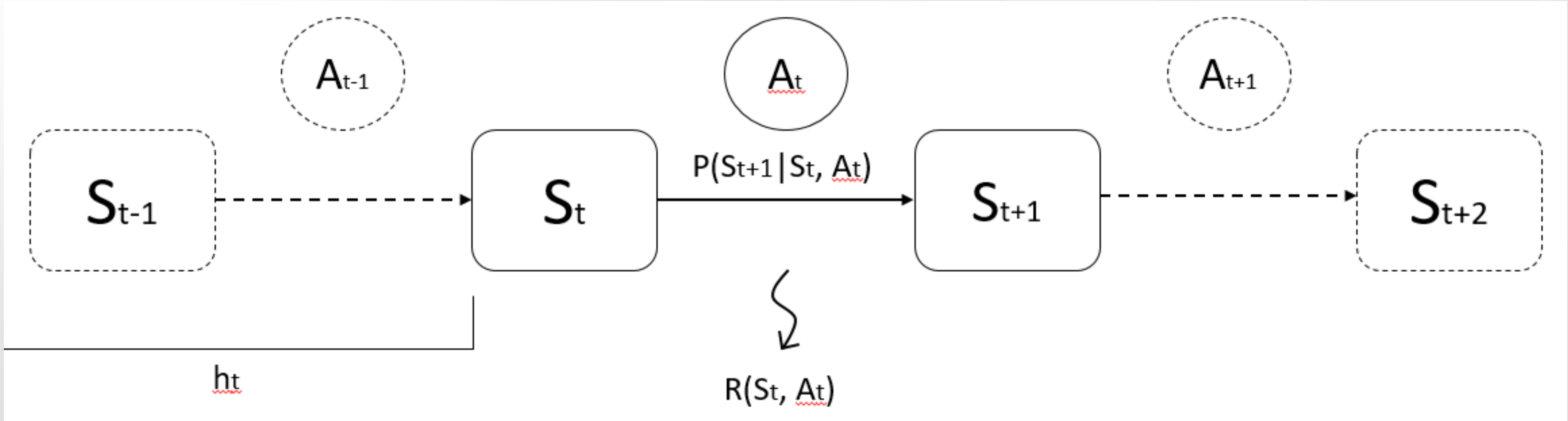


<u>Informations :</u>

- List of possible actions
- Reward :
  - Positive
  - Negative
  - Null
- Value

How does agent choose his action ?

Markov Decision Process



- S : sates set
- A : actions set
- T : time
- P() : probability of transition between states
- R() : reward function for transition
- h : historic of actions

How does agent choose his action ?

The policy

Definition :

Procedure to be followed by the agent to choose at a moment the action to be executed (noted π)

GOAL → found the optimal policy (π*)

Bellman equation

Estimation of the future reward if the agent do a action

$$V^{\pi}(s) \quad = \quad \sum_{a \in \mathcal{A}(s)} \pi(s,a) \sum_{s' \in \mathcal{S}} \mathcal{P}(s,a,s')[\mathcal{R}(s,a,s') + \gamma V^{\pi}(s')]$$

P(s,a,s') : probability to go to the state s' with the action a if I am in s

[1] Sutton, R. Planning by incremental dynamic programming.
In *Eight International Workshop on Machine Learning*, pages 353-357. Morgan Kaufmann. 1991
[2] L.P. Kaelbling, M. L. Littman, A.W. Moore, « Reinforcement Learning : A Survey », Journal of Artificial Intelligence Research 4, 1996
[3] S. Russell P. Norving, Book « Artifial Intelligence : A Modern Approach » 2010

Reinforcement learning algorithm : Q-Learning

The value of the estimation is not on the state S_t but on the action to go from S_t to S_{t+1}

From Bellman to Q-Learning

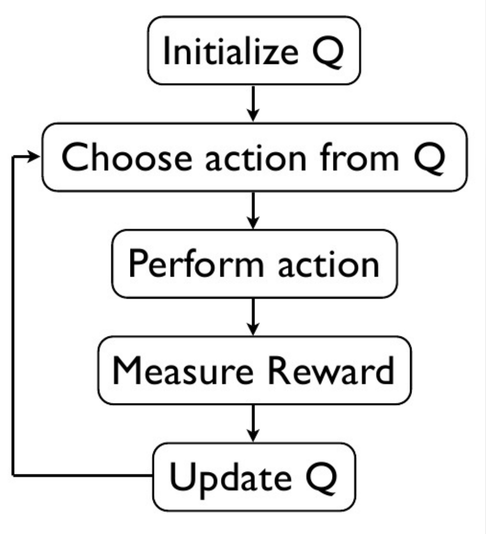
$$V^{\pi}(s) = \sum_{a \in \mathcal{A}(s)} \pi(s,a) \sum_{s' \in \mathcal{S}} \mathcal{P}(s,a,s')[\mathcal{R}(s,a,s') + \gamma V^{\pi}(s')]$$

$$Q^{*}(s,a)=\sum P(s,a,s')[R(s,a,s')+\gamma \max_{a'} Q^{*}(s',a')]$$

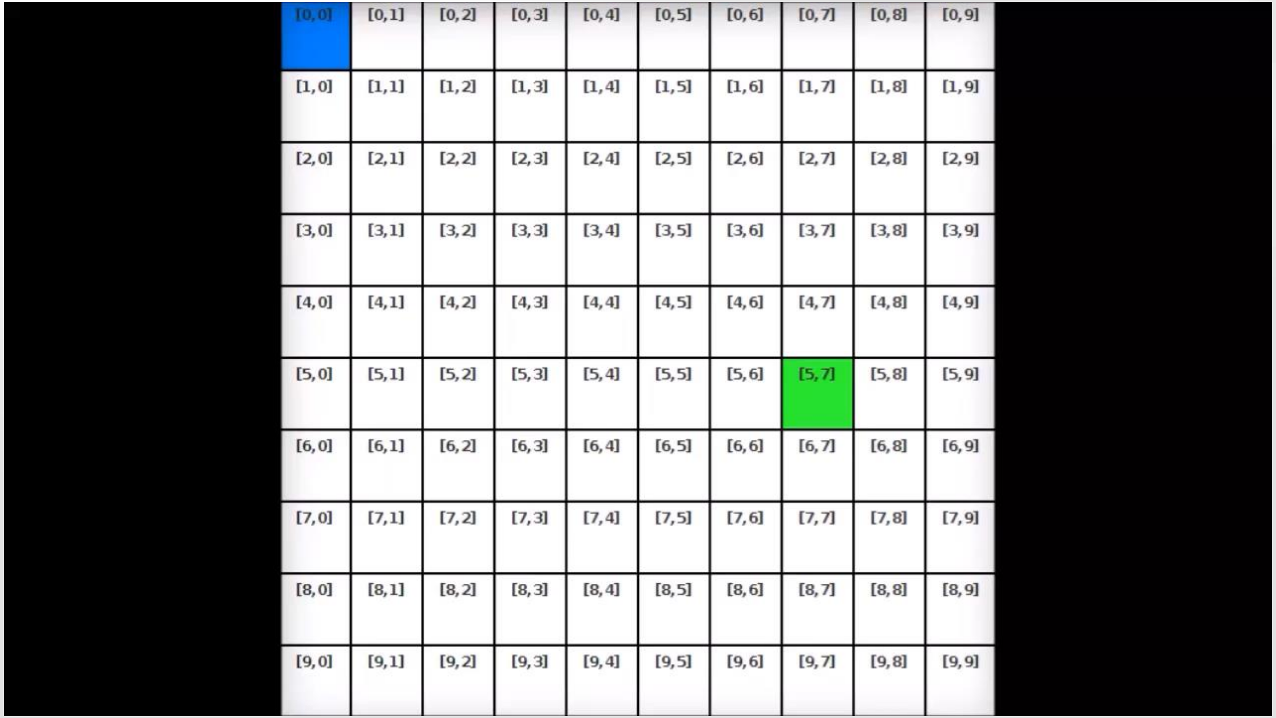
Actualization function

$$\boxed{Q(s_t,a_t)=Q(s_t,a_t)+\alpha(R_t+\gamma \max_{a} Q(s_{(t+1)},a_{(t+1)})-Q(s_t,a_t))}$$

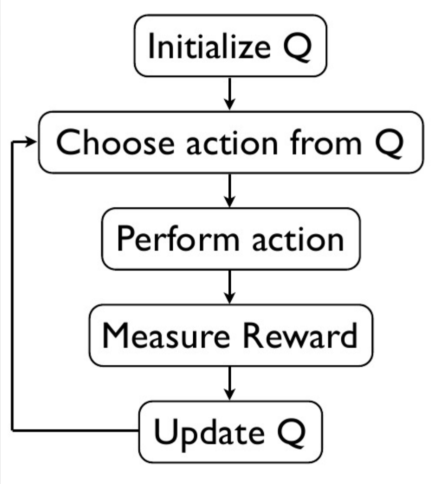
Algorithm



Q-Learning, algorithm explication



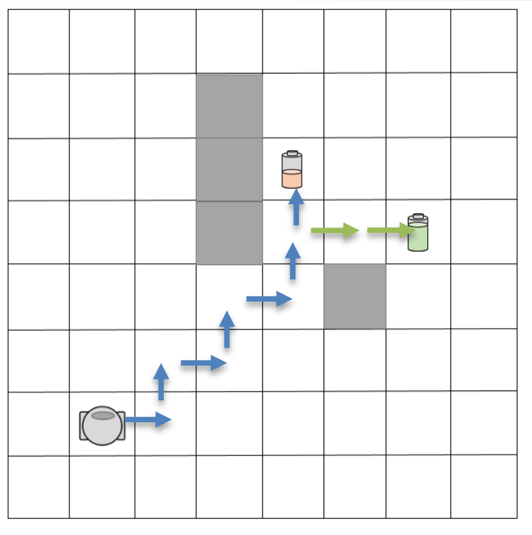
$$Q(s_t,a_t)=Q(s_t,a_t)+\alpha\left(R_t+\gamma\cdot\max_a Q(s_{(t+1)},a_{(t+1)})-Q(s_t,a_t)\right)$$



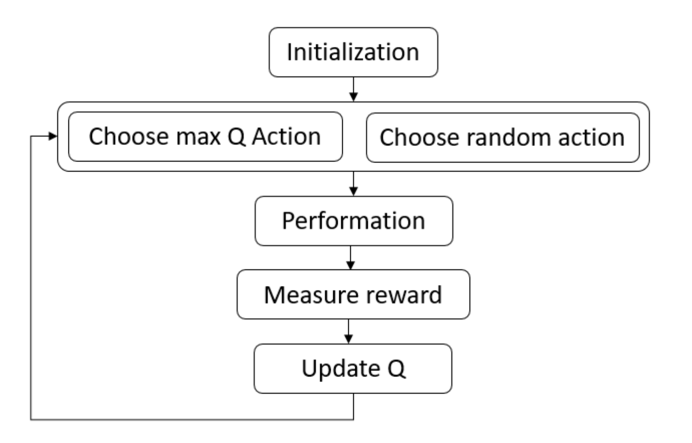
1920 moves 19 backup 2mins (16moves/sec) α = 0,5 γ = 0,5

Issue exploration vs exploitation

Issue

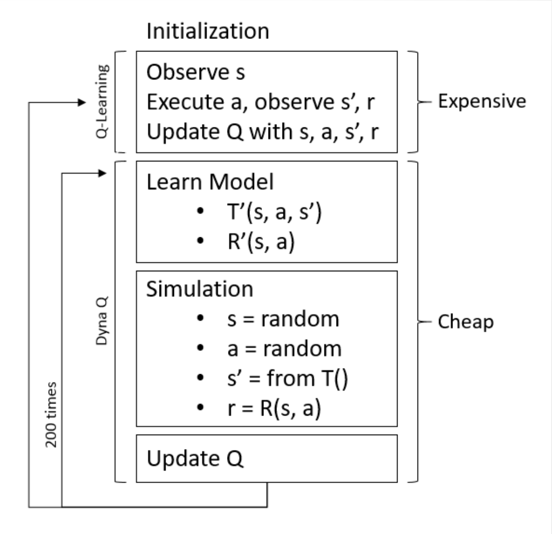


Algorithm

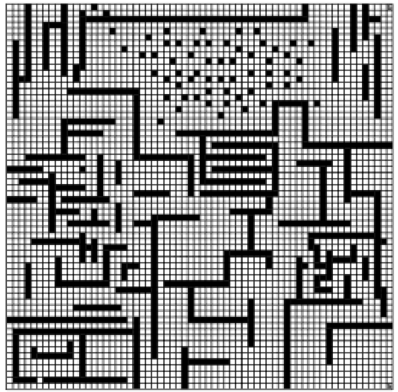


Possible amelioration : Dyna Q

Algorithm



Test



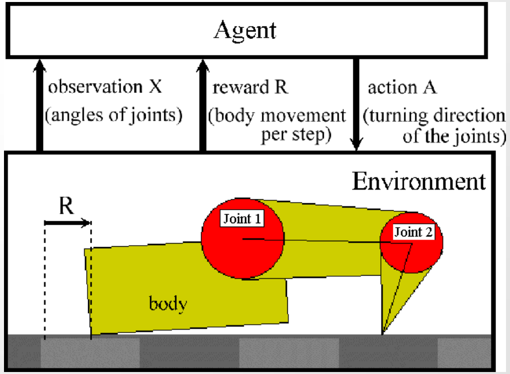
	Steps before convergence	Backups before convergence
Q-learning	531,000	531,000
Dyna	62,000	3,055,000
prioritized sweeping	28,000	1,010,000

[1] L.P. Kaelbling, M. L. Littman, A.W. Moore, « Reinforcement Learning : A Survey », Journal of Artificial Intelligence Research 4, 1996
[2] Watkins, C.J.C.H. “Q-Learning”, Machine Learning, 8(3), 279-292 , 1992
[3] H. Larochelle, YouTube video « Intelligence Artificielle [13.7] : Apprentissage par renforcement – Q-learning »
[4] P.Norving, S.Thrun, Udacity cours, « Introduction to Artificial Inteligence » 2010

Personnal implemantation

In this project we will simulate the evolution of a population in a urban environment

Reinforcement learning in robotics



To many states ⇒ exploration time
Delimitate states in real world

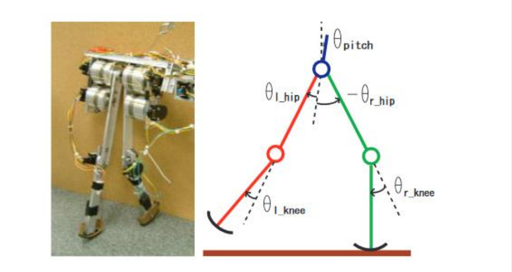
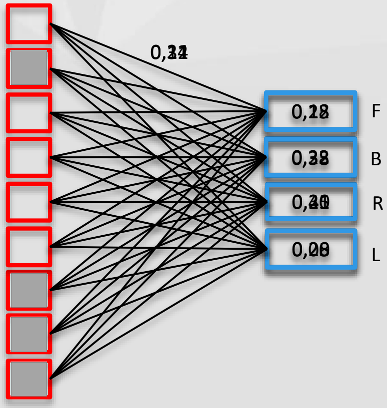
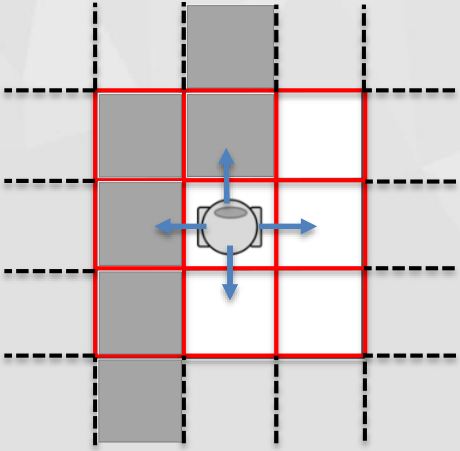


Fig. 2. Five link biped robot

Function approximator





- Make a generalization
- Use only the visible states

Issues, discussion

Lemma : let n denote the number of actions applicable at state s'. If all n actions share target Q-Value, i.e, $\exists q \,:\, \forall \hat{a} : q = Q^{target}(s', \hat{a}), \mathit{then\ the\ average\ overestimation}\ E[Z_s] \mathit{is}\ \gamma c\ \mathit{with}\ c = \ \varepsilon \frac{n-1}{n+1}$

Table 1: Upper bound on the error ε of the function approximator, according to Theorem 2. These bounds are significant. For example, if episodes of length $L = 60$ with $n = 5$ actions shall be learned, ε must be smaller than .00943 (bold number).

	L=10	L=20	L=30	L=40	L=50	L=60	L=70	L=80	L=90	L=100	L=1000
$n = 2$	.12991	.05966	.03872	.02866	.02275	.01886	.01611	.01405	.01247	.01120	.00110
$n = 3$	.08660	.03977	.02581	.01911	.01517	.01257	.01074	.00937	.00831	.00746	.00073
$n = 4$	.07217	.03314	.02151	.01592	.01264	.01048	.00895	.00781	.00692	.00622	.00061
$n = 5$	.06495	.02983	.01936	.01433	.01137	**.00943**	.00805	.00702	.00623	.00560	.00055
$n = 6$	.06062	.02784	.01807	.01337	.01061	.00880	.00751	.00656	.00581	.00522	.00051
$n = 8$	.05567	.02557	.01659	.01228	.00975	.00808	.00690	.00602	.00534	.00480	.00047
$n = 10$	.05292	.02430	.01577	.01167	.00927	.00768	.00656	.00572	.00508	.00456	.00045
$n = 20$	.04786	.02198	.01426	.01056	.00838	.00695	.00593	.00517	.00459	.00412	.00040
$n = \infty$	.04330	.01988	.01290	.00955	.00758	.00628	.00537	.00468	.00415	.00373	.00036

[1] J.Morimoto, G.Cheng, C.G.Atkeson, G.Zeglin, « A simple reinforcement learning algorithm for biped walking »
International conference on robotics & automation, New Orleans, 2004
[2] S.Thrun, A.Schwartz « Issues in using function approximation for reinforcement learning »
Proceedings of the Fourth Connectionist Models Summer School, Lawrence Erlbaum Publisher, Hillsdale, NJ, Dec. 1993
[3] L.P. Kaelbling, M. L. Littman, A.W. Moore, « Reinforcement Learning : A Survey », Journal of Artificial Intelligence Research 4, 1996
[4] S. Russell P. Norving, Book « Artifial Intelligence : A Modern Approach » 2010