

TP « ALPHA1 » 1 – Découverte de l'Intelligence Artificielle avec le robot Alpha1

Ce TP a pour objectif de faire comprendre comment une machine (un robot, ou un programme d'une manière plus général) est capable d'apprendre, et plus particulièrement de visualiser l'apprentissage par un réseau de neurones artificiels.

I. Créer soi-même des connexions pour programmer le comportement du robot.

Dans cette première partie nous allons voir comment un mini réseau de neurones artificiels permet de prendre des décisions d'action en fonction des données capteurs. Pour l'instant il n'y a pas d'apprentissage automatique : c'est *vous* qui allez créer les connexions appropriées à l'intérieur du réseau de neurones, pour obtenir le maximum de récompenses.



OBJECTIF : Créer les bonnes connexions pour maximiser les récompenses et le niveau du robot.

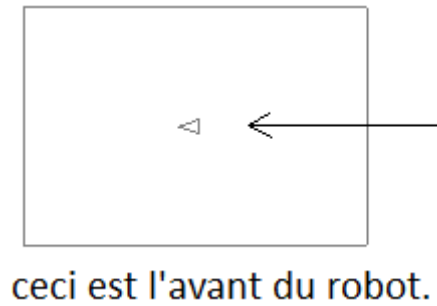
ETAPES :

1. Lancer le programme : vous observez à gauche la partie contrôle, à droite la partie visualisation

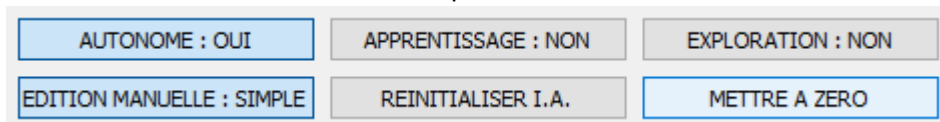
Alpha1, l'Education à l'Intelligence Artificielle

Récompense 0 Niveau 0

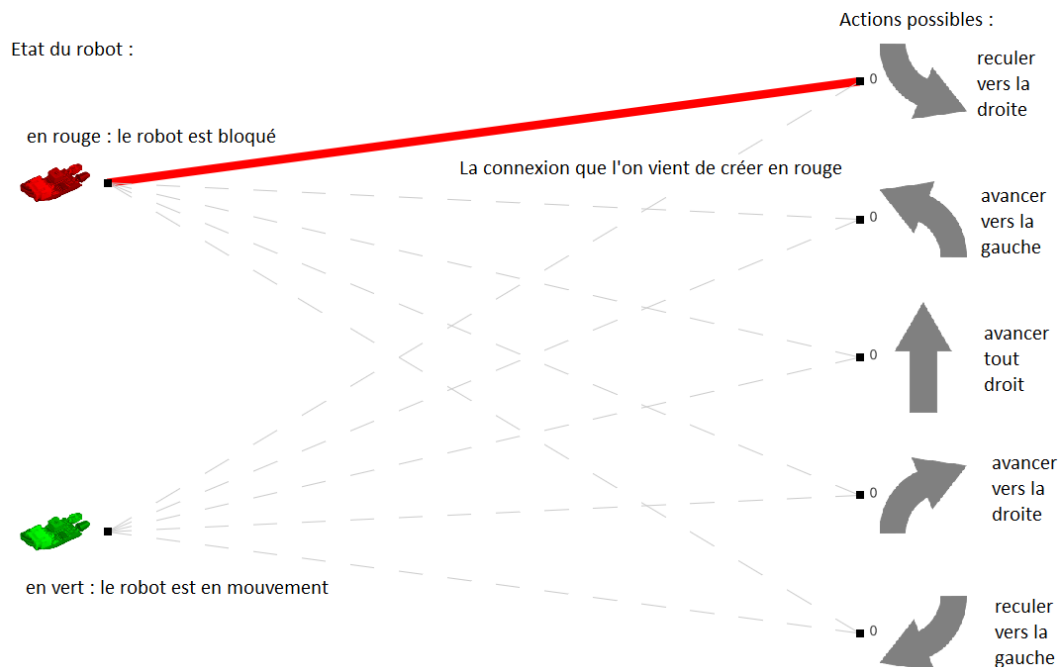
2. Dans un premier temps, vous allez commencer par utiliser le simulateur du robot, vérifiez que la case **simulation** soit bien cochée. Vous pourrez observer alors le mini robot simulé dans l'encadré de l'interface :



3. C'est vous qui allez créer les connexions : Désactivez « APPRENTISSAGE » et « EXPLORATION », activez « EDITION MANUELLE » et cliquez « METTRE A ZERO »



4. Vous avez devant vous une représentation des connexions dans le réseau de neurones artificiels. Ce réseau a 7 neurones : 2 neurones d'entrée et 5 neurones d'action.



5. Si vous survolez avec la souris les connexions en pointillés, vous pourrez alors cliquer afin de **créer une nouvelle connexion**. Ces connexions vont transmettre l'activité des neurones d'entrée vers les sorties ; elles correspondent donc à la sélection d'une action (sur la droite) lorsque le robot se trouve dans un état en particulier (sur la gauche). Dans l'exemple, la connexion créée, va permettre au robot de choisir de **reculer vers la droite** lorsqu'il **est bloqué**.
6. Désormais, à vous de jouer, créer les bonnes connexions pour permettre au robot **de récupérer le maximum de récompenses**. Un indice : les récompenses sont calculées comme la vitesse en avant du robot...
7. Vous êtes prêt ? Appuyez sur le bouton **DEMARRER** et observez les **récompenses** qu'obtient le robot et comment évolue son **niveau** (le niveau est la moyenne des récompenses durant les 2 dernières minutes).



Essayez de modifier les connexions pour peut-être en trouver de meilleures...

8. Appeler nous pour vérifier que vous avez créé les bonnes connexions !
9. Ensuite c'est le moment d'essayer avec le vrai robot ! Pour se connecter au robot, il faut :
 - Allumer le robot, et attendre qu'il clignote en blanc.
 - Se connecter au réseau Wi-Fi du robot : AlphaAIXXXX. Le XXXX correspond au numéro de votre robot, se numéro est gravé en dessous du robot. (Le mot de passe : OmegaBot).
 - Décocher la case simulation

Le robot se comporte-t-il comme prévu ? Quel niveau atteint-il ?

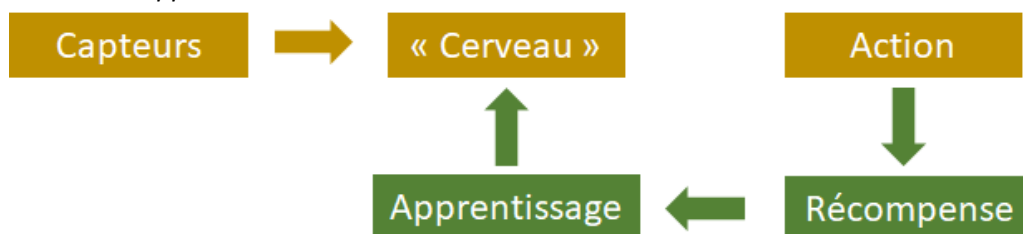
II. Observer comment le robot apprend par lui-même.

Maintenant que vous avez compris comment le robot a la possibilité de « choisir » ses actions (grâce aux connexions), nous allons voir comment le robot est capable d'apprendre quelles sont les bonnes actions à choisir ! (toujours dans l'objectif de maximiser ses récompenses)

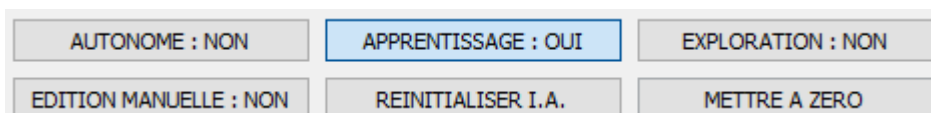
OBJECTIF : Activer les bons paramètres pour permettre au robot d'apprendre quelles sont les meilleures actions.

ETAPES « ROBOT PILOTÉ » :

1. Pour les étapes qui suivent, à vous de choisir si vous voulez les faire en utilisant le vrai robot ou le robot simulé (vous pouvez bien sûr commencer avec le simulé puis refaire avec le vrai robot).
2. Désactivez « EDITION MANUELLE » et à la place activez « APPRENTISSAGE ». Cette fois cliquez « REINITIALISER I.A. » : cela met des connexions au hasard (remarquez-vous qu'elle ne sont pas très bonne ? elles vont *s'améliorer* au cours de l'apprentissage)
3. Pour commencer c'est *vous* qui allez piloter le robot, ce n'est pas lui qui va choisir ses actions, il va seulement *apprendre*.



Pour cela désactivez « AUTONOME ».



4. Cliquez **DEMARRER**. Le robot attend que vous le pilotiez ! Vous pouvez le faire en cliquant sur les

flèches de l'interface ou en utilisant les flèches du clavier .

5. Observez attentivement l'épaisseur des connexions et les chiffres qui se situent à côté des actions possibles ; à votre avis, à quoi sert ce chiffre ? que signifie-t-il ? pourquoi évolue-t-il ? comment évolue-t-il ? Quel peut être le calcul qui est derrière ?

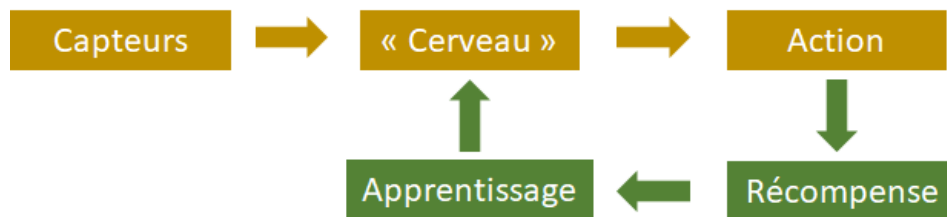
17



Appelez-nous pour nous faire part de vos réflexions ! Un indice : il indique le résultat de l'**apprentissage**...

6. Lorsque vous pensez que l'apprentissage est terminé, désactivez « APPRENTISSAGE » pour l'arrêter et réactivez « AUTONOME » pour que le robot se conduise tout seul : se comporte-t-il comme attendu ? Si ce n'est pas le cas, réactivez l'apprentissage pour le faire apprendre un peu mieux...

ETAPES « ROBOT AUTONOME » :



7. Recommencez l'apprentissage cette fois-ci avec le robot qui choisit ses actions par lui-même (appuyez sur « REINITIALISER I.A. », vérifiez que « AUTONOME » est activé, cliquez **DEMARRER**). Qu'observez-vous ? Le robot apprend-il correctement ? Que peut-il manquer ?
8. A présent activez également « **EXPLORATION** » et observez comment se passe l'apprentissage. Que fait ce paramètre ? Appelez-nous pour nous faire part de vos réflexions.

COMPRENDRE TOUS LES DETAILS MATHEMATIQUES DE L'APPRENTISSAGE :

9. Vous avez observé que l'apprentissage modifie les connexions dans le réseau de neurone. Continuez les étapes suivantes pour en comprendre tous les détails et arriver à l'équation . Vous pouvez également sauter directement à la partie III pour commencer les apprentissages avec caméra et revenir à cette étape à la fin s'il vous reste du temps.
10. Affichez l'onglet « I.A. » : vous voyez apparaître 3 paramètres, *vitesse d'apprentissage*, *gamma* et *exploration*.

11. Le paramètre *exploration* représente la fréquence des explorations : variez-le entre 0 et 1 et observez l'effet sur le comportement du robot. Vous semble-t-il que la valeur initiale de 0.1 était correcte ?
12. A présent revenons à quelque chose de plus simple : mettez vitesse d'apprentissage à 1 et gamma à 0.

vitesse d'apprentissage	<input type="text" value="1.0"/>
gamma	<input type="text" value="0.0"/>
exploration	<input type="text" value="0.1"/>

Cela revient après qu'on a fait une action a_t à partir de l'état s_t à remplacer la connexion entre s_t et a_t par la récompense reçue r_{t+1} .

$$Q(s_t, a_t) = r_{t+1}$$

Essayez un apprentissage avec ces valeurs des paramètres. Que se passe-t-il ?

13. Pour éviter des fluctuations trop forte des connexions sans jamais converger vers une valeur stable, diminuez la vitesse d'apprentissage. L'équation des apprentissages devient alors (α est le paramètre "vitesse d'apprentissage").

$$Q(s_t, a_t) += \alpha(r_{t+1} - Q(s_t, a_t))$$

Obtient-on de meilleurs apprentissages ? Voyez-vous quelque chose qui manque encore ?

14. Pour permettre au réseau de neurones d'apprendre que quand le robot est bloqué l'action « se retourner » est meilleure que les actions « tout droit » ou « tourner », il faut intégrer dans la valeur des actions la qualité du nouvel état dans lequel on arrive (ici, bloqué ou non bloqué). C'est ce que permet de faire le paramètre gamma. L'équation complète de l'apprentissage devient :

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

Appelez-nous pour nous demander de vous aider à la comprendre !

III. Ajouter la caméra pour un comportement plus complexe.

Maintenant que vous savez comment le robot adapte ses connexions pour aller choisir les actions les plus pertinentes, nous allons ajouter une caméra au robot pour qu'il puisse apprendre à anticiper les obstacles !

Mais attention, avant de pouvoir utiliser la caméra, il va falloir d'abord découvrir de nouveaux paramètres !

OBJECTIF : Etablir les paramètres corrects pour de bons apprentissages avec la caméra.

ETAPES « MEMOIRE D'EXPERIENCE » :

1. Dans l'onglet « I.A. » changez l'algorithme pour mettre « deep Q-learning », cela va permettre d'utiliser des réseaux de neurones multi-couches et d'activer de nouveaux paramètres.

Robot	I.A.	Visualisation
V Apprentissage		
algorithme	<input type="text" value="deep Q-learning"/>	
vitesse d'apprentissage	<input type="text" value="0.2"/>	
gamma	<input type="text" value="0.8"/>	
exploration	<input type="text" value="0.1"/>	
mémoire d'expériences	<input type="checkbox"/>	
V Réseau de neurones		
couches de neurones intermédiaires	<input type="text"/>	
biais neuronal	<input checked="" type="checkbox"/>	

Lancez un nouvel apprentissage : vous constaterez que les connexions et les valeurs des actions peuvent cette fois être positives *ou négatives*. Vérifiez qu'un premier apprentissage se déroule bien.

2. Activez l'option « mémoire d'expériences » et relancez l'apprentissage. Normalement il devrait être plus rapide !! En effet l'apprentissage est très accéléré car les connexions sont modifiées de nombreuses fois par seconde, non pas en fonction de la dernière action réalisée seulement, mais *de toutes les actions effectuées et récompenses reçues* jusqu'à présent. Il suffit donc de faire une exploration tout droit pour apprendre que aller tout droit est meilleur que tourner à gauche ou à droite.

ETAPES « NEURONES INTERMEDIAIRES » :

3. Ajoutez une couche de neurones intermédiaires et regardez ces nouveaux neurones et de nouvelles connexions apparaître dans l'interface !

V Réseau de neurones

couches de neurones intermédiaires

biais neuronal

Relancez l'apprentissage. Comme précédemment l'algorithme d'apprentissage apprend les valeurs correctes des actions, mais le calcul de ces valeurs devient beaucoup plus complexe puisqu'il dépend d'un grand nombre de connexions. Appelez-nous pour en discuter avec nous.

ETAPES « CAMERA » :

4. Enfin ! Vous pouvez activer la caméra ; choisissez la taille 16x12 ou 32x24.

Robot	I.A.	Visualisation
simulation	<input checked="" type="checkbox"/>	
> Vitesse		
V Capteurs		
vitesse	-	▼
suivi de ligne	bloqué / en mouvement	▼
RSSI Bluetooth	-	▼
ultra-son	-	▼
caméra	in_v_16x12	▼

Et voici les réglages que nous vous conseillons pour l'IA : vous pourrez aussi en essayer d'autres !

Robot	I.A.	Visualisation
V Apprentissage		
algorithme	deep Q-learning	▼
vitesse d'apprentissage	0.05	<input type="text"/>
gamma	0.8	<input type="text"/>
exploration	0.1	<input type="text"/>
mémoire d'expériences	<input checked="" type="checkbox"/>	
taille mémoire	1000	<input type="text"/>
V Réseau de neurones		
couches de neurones intermédiaires	100 100 50	<input type="text"/>
biais neuronal	<input checked="" type="checkbox"/>	

- Lancez avec **DEMARRER** et observez l'apprentissage du robot : normalement il apprend successivement à aller tout droit, à se retourner quand il cogne dans un obstacle, puis à reconnaître ces obstacles à l'avance et tourner avant de cogner dedans ! Cet apprentissage prend à peu près 10 minutes. Jouez avec les options de visualisation pour voir séparément les connexions dans le réseau de neurone, ses activités, ses apprentissages...

IV. Faire jouer AlphaI au foot !

Vous vous rappelez que le but du robot est d'obtenir le maximum de **récompenses**. Si nous modifions comment ces récompenses sont calculées, nous pouvons lui faire apprendre quelque chose de tout à fait différent !!!!

OBJECTIF : Faire apprendre au robot à pousser un ballon vert.

ETAPES :

- Gardez les paramètres qui ont marché pour l'apprentissage précédent avec la caméra, et ne modifiez que le mode de calcul de la récompense !! Voici les réglages pour récompenser le robot lorsqu'il avance vers le ballon vert : le principe est qu'on détecte combien de pixels sont vert dans l'image de la caméra.

The screenshot shows the 'Récompense' (Reward) settings panel. The 'mode de récompense' (reward mode) is set to 'couleur dans l'image, le robot ne doit' (color in the image, the robot must not). Below this, several parameters are set in input fields:

restriction de la zone de l'image récompensée	0.0 0.0
teinte minimale	1.3
teinte maximale	2.5
luminance minimale	0.3
luminance maximale	0.8
saturation minimale	0.4

- Pour voir les pixels récompensés à l'écran, cocher « pixels de couleur récompensée »

The screenshot shows the 'Caméra' (Camera) settings panel. The 'pixels de couleur récompensée' (rewarded color pixels) checkbox is checked. Other settings include 'résolution dégradée' (degraded resolution) checked and 'colorier les neurones "pixels"' (color neurons "pixels") checked.

- Il sera utile également de diminuer « temps boucle » pour rendre le robot plus réactif (avec 0.25 il prendra 4 décisions par seconde au lieu de 2)

The screenshot shows the 'Vitesse' (Speed) settings panel. The 'temps boucle' (loop time) is set to 0.25.

4. Vous êtes prêt ! Mettez un ballon vert dans l'arène et lancez l'apprentissage ! Cet apprentissage prend un peu plus de temps, le robot apprend d'abord à suivre le ballon lorsque celui-ci est proche, il met plus de temps à le reconnaître de loin.
5. Il peut arriver que les pixels verts dans l'image ne soient pas bien détectés. Dans ce cas :
 - a. stoppez le robot,
 - b. réglez les « **actions possibles** » dans les paramètres à « **stop** ».
 - c. Positionnez le robot face au ballon vert.
 - d. Démarrer le robot ! (Le robot ne va pas bouger cependant)
 - e. Appelez-nous pour vous aider à régler les paramètres pour la récompense de la couleur
6. Vous êtes arrivés ici ? Vous pouvez relancer de nouveaux apprentissages en modifiant les paramètres que vous avez appris !!!

V. Conclusion

Récapitulons quelques notions que vous avez apprises :

- Ce qu'on appelle « Intelligence Artificielle » est en fait mal défini, plus qu'une *méthode*, c'est un *projet* : celui de reproduire l'intelligence biologique !!!
- Les grands progrès récents concernent l'*Apprentissage Machine* (en anglais, *Machine Learning*) : nous avons découvert cela de près aujourd'hui !
- La méthode souvent la plus efficace est l'utilisation de *réseaux de neurones artificiels* qui copient l'activité et les apprentissages des neurones dans notre propre cerveau !! Plus précisément il s'agit d'augmenter et diminuer les bonnes *connexions* entre les neurones artificiels.
- Par conséquent certaines idées de ces algorithmes renvoient à nos propres apprentissages ! Essais et erreurs ; Curiosité ; Temps d'apprentissage, Répétition et Apprentissage pendant le repos ; etc.
- Tout cela s'appuie sur des équations mathématiques rigoureuses. Les chercheurs en Intelligence Artificielle doivent être particulièrement qualifiés en *statistiques*, mais aussi en *programmation* et en ayant *un bon sens intuitif*.