Utilisation de l'intelligence artificielle dans la manœuvre autonome de bateau

Maxime CAUTRÈS

Lycée Blaise Pascal

01/03/2020

Sommaire

Données économiques

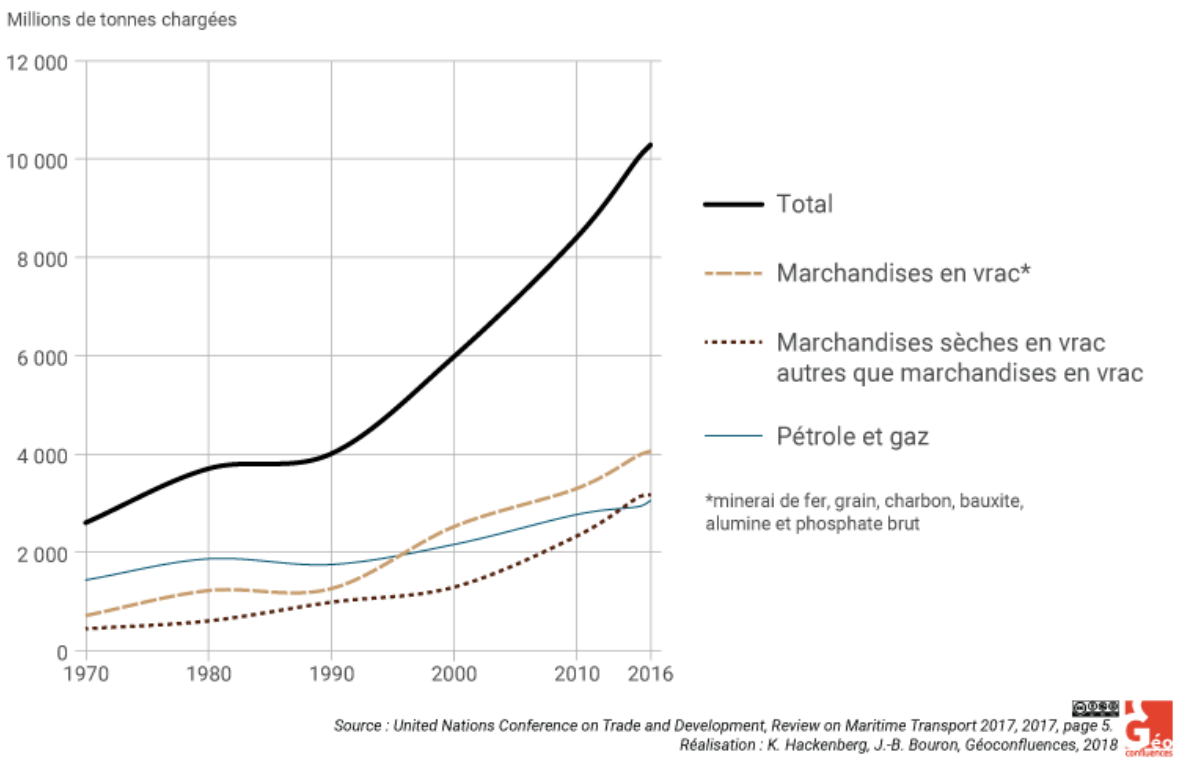


FIGURE – La croissance du commerce maritime international (en millions de tonnes chargées) [1]

1. http ://geoconfluences.ens-lyon.fr/informations-scientifiques/dossiers-regionaux/territoires-europeens-regions-etats-union/rte-t/port-anvers

Le métier de pilote maritime

FIGURE – Transfert du pilote maritime sur le bateau à piloter [3]

- Un métier **dangereux** (Le transfère du pilote)

3. http ://escale.sinerj.org/spip.php ?article41

Le métier de pilote maritime

FIGURE – Transfert du pilote maritime sur le bateau à piloter [3]

- Un métier **dangereux** (Le transfère du pilote)
- Un **coût matériel** important (Bateau ou hélicoptère)

3. http ://escale.sinerj.org/spip.php ?article41

Le métier de pilote maritime

FIGURE – Transfert du pilote maritime sur le bateau à piloter [3]

- Un métier **dangereux** (Le transfère du pilote)
- Un **coût matériel** important (Bateau ou hélicoptère)
- Un **coût financier** important (7% du coût de l'escale)

3. http ://escale.sinerj.org/spip.php ?article41

Étude de l'existant

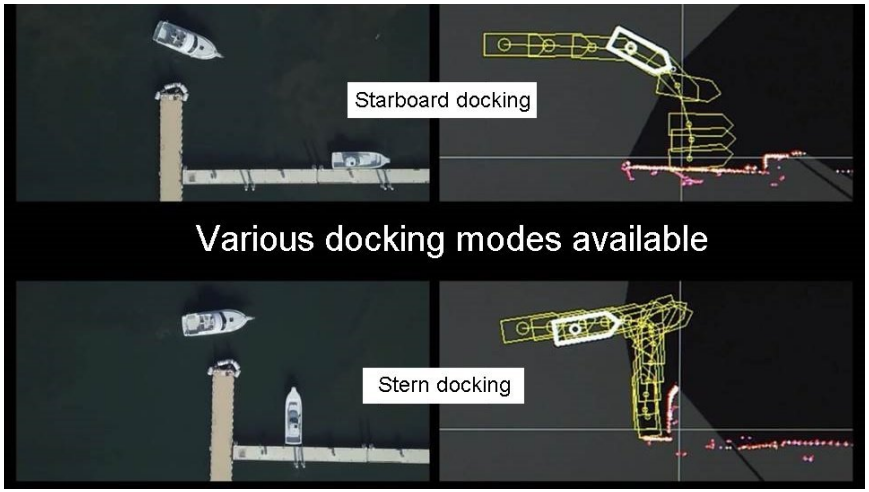


FIGURE – Vu aérienne de la trajectoire suivit par l'asservissement du bateau [5]

- **Peu d'acteurs** dans le domaine (Deux principaux avec Yanmar et Volvo)

5. https ://smartmaritimenetwork.com/2019/02/08/yanmar-trials-robotic-ship-technology/

Étude de l'existant

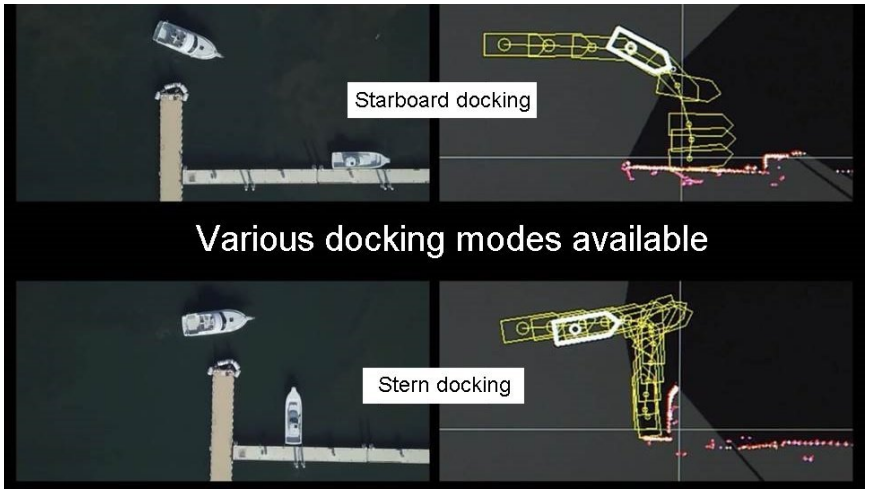


FIGURE – Vu aérienne de la trajectoire suivit par l'asservissement du bateau [5]

- **Peu d'acteurs** dans le domaine (Deux principaux avec Yanmar et Volvo)
- Nécessite des modifications importantes des **infrastructures** (capteurs, antennes)

5. https ://smartmaritimenetwork.com/2019/02/08/yanmar-trials-robotic-ship-technology/

Étude de l'existant

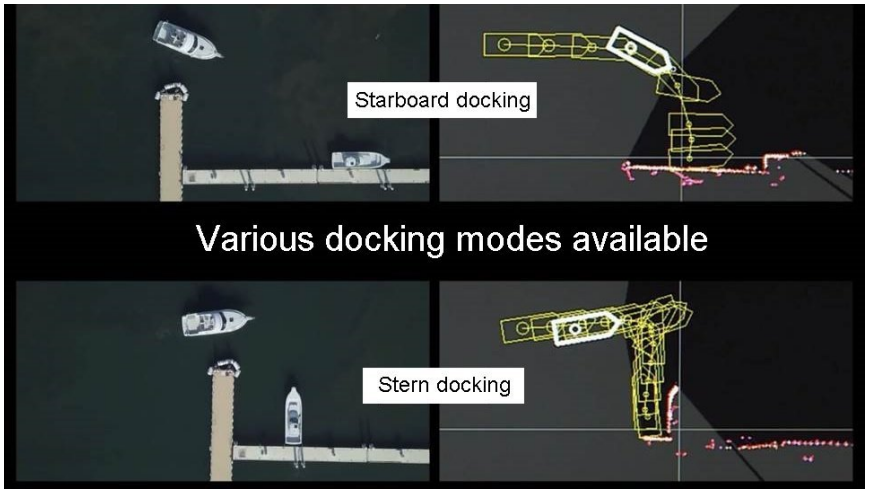


FIGURE – Vu aérienne de la trajectoire suivit par l'asservissement du bateau [5]

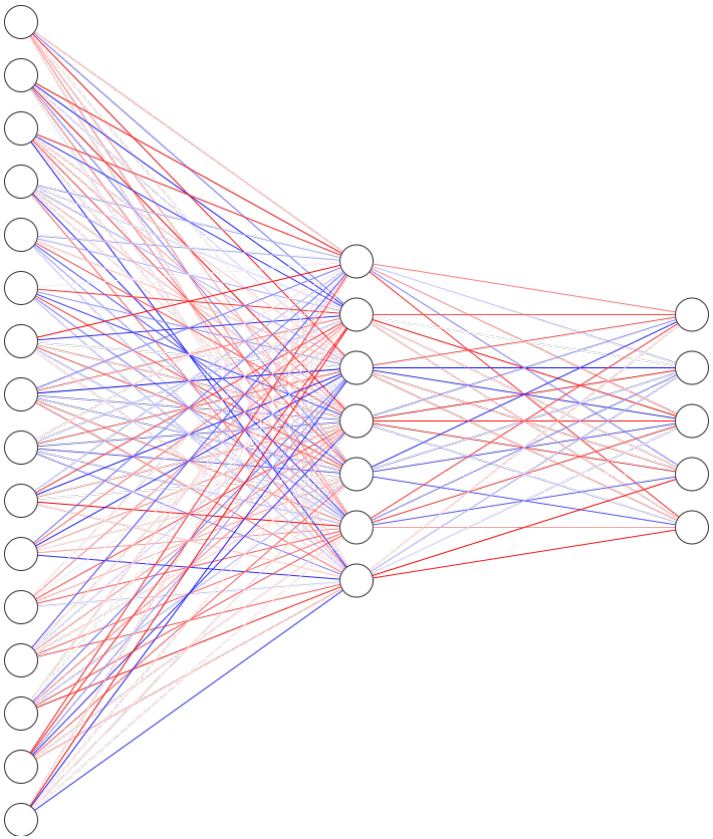
- **Peu d'acteurs** dans le domaine (Deux principaux avec Yanmar et Volvo)
- Nécessite des modifications importantes des **infrastructures** (capteurs, antennes)
- Un dispositif **très lent et peu adapté** aux déplacements important dans un port

5. https ://smartmaritimenetwork.com/2019/02/08/yanmar-trials-robotic-ship-technology/

L'apprentissage automatique :

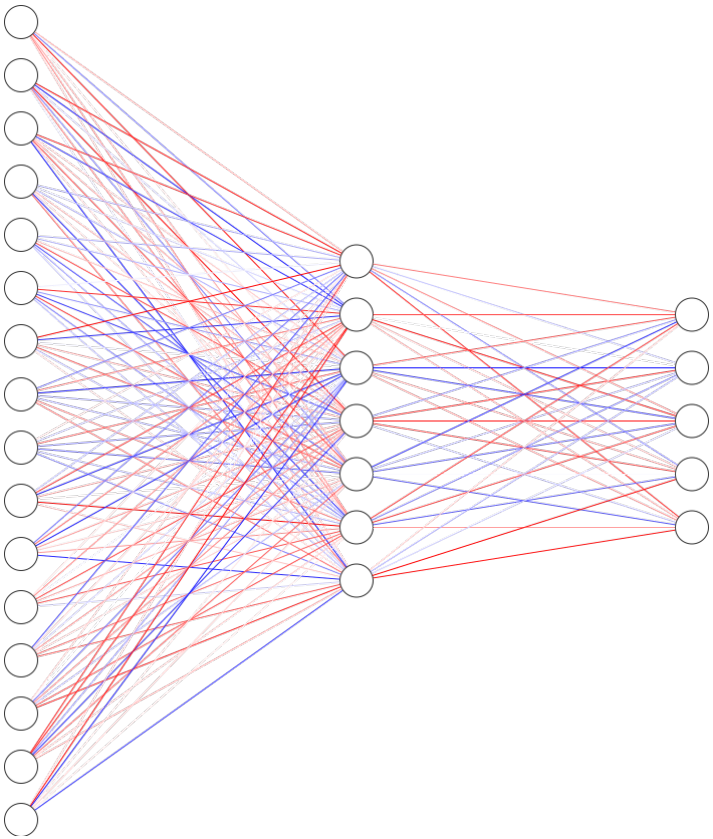
- Un environnement pour simuler les conditions réelles

L'apprentissage automatique :



- Un environnement
- La techonologie des réseaux de neurones

L'apprentissage automatique :



- Un environnement
- La techonologie des réseaux de neurones
- Des algorithmes d'entrainement

Problématique

Comment peut-on utiliser l'**apprentissage automatique** pour permettre à un bateau de **manœuvrer dans un port** dans le but de minimiser les dépenses liées à l'augmentation du trafic tout en garantissant la sécurité ?

Problématique

Comment peut-on utiliser l'**apprentissage automatique** pour permettre à un bateau de **manœuvrer dans un port** dans le but de minimiser les dépenses liées à l'augmentation du trafic tout en garantissant la sécurité ?

Le plan :

Problématique

Comment peut-on utiliser l'**apprentissage automatique** pour permettre à un bateau de **manœuvrer dans un port** dans le but de minimiser les dépenses liées à l'augmentation du trafic tout en garantissant la sécurité ?

Le plan :

- Première approche avec le Q-learning

Problématique

Comment peut-on utiliser l'**apprentissage automatique** pour permettre à un bateau de **manœuvrer dans un port** dans le but de minimiser les dépenses liées à l'augmentation du trafic tout en garantissant la sécurité ?

Le plan :

- Première approche avec le Q-learning
- Simulation de l'environnement portuaire

Problématique

Comment peut-on utiliser l'**apprentissage automatique** pour permettre à un bateau de **manœuvrer dans un port** dans le but de minimiser les dépenses liées à l'augmentation du trafic tout en garantissant la sécurité ?

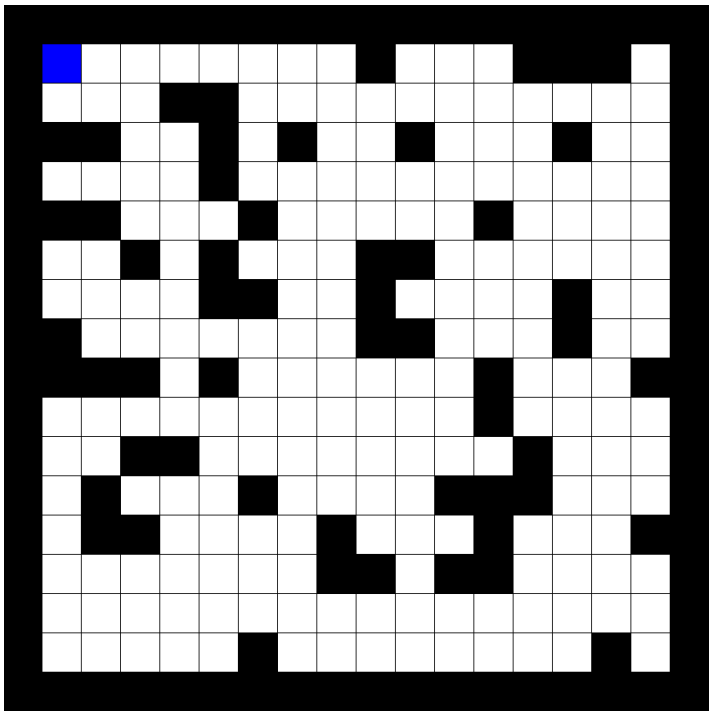
Le plan :

- Première approche avec le Q-learning
- Simulation de l'environnement portuaire
- Seconde approche avec le Policy Gradients

Sommaire

Un problème intermédiaire pour se lancer

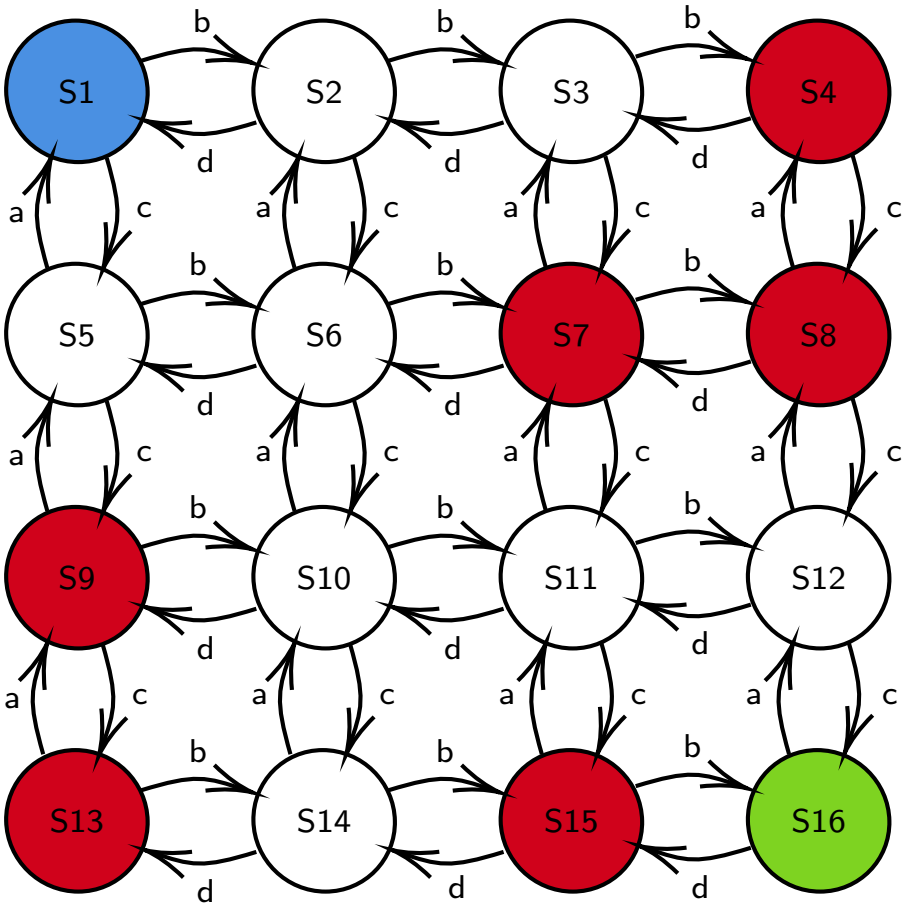
Description



- En Noir les obstacles
- En blanc les cases accessibles
- La souris est en bleu
- L'objectif en la case en bas a droite

Un problème intermédiaire pour se lancer

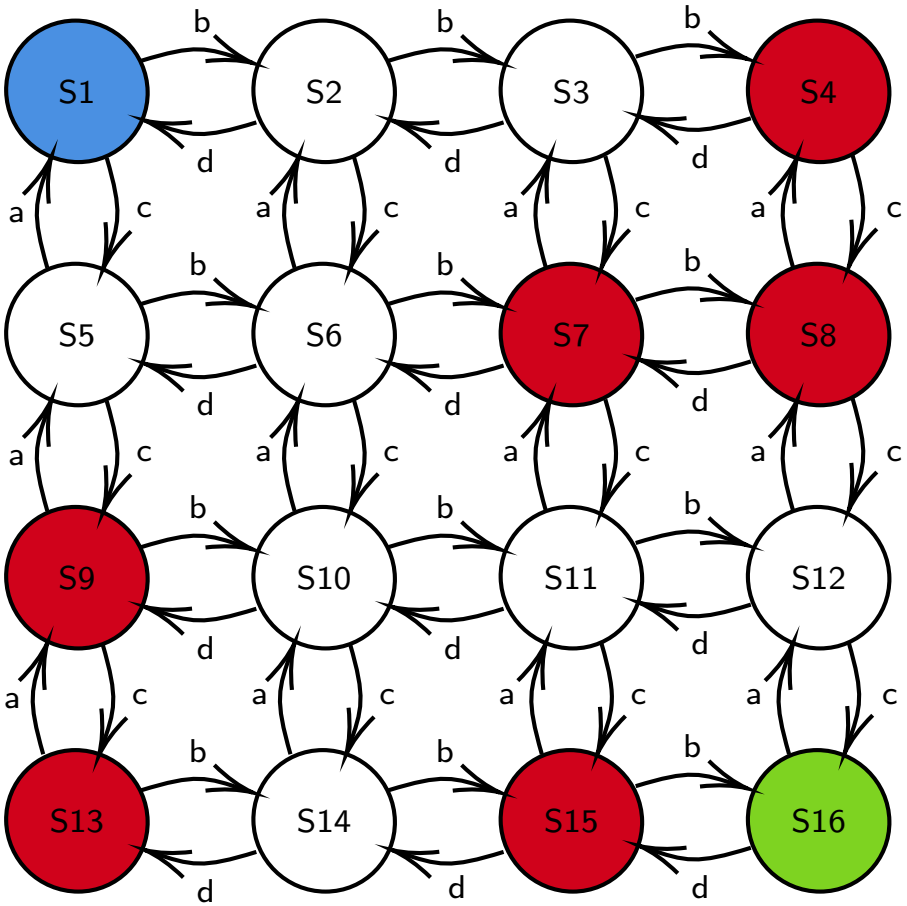
Formalisation



- On utilise **les chaines de Markov deterministe**
- Bleu pour l'état initial
- Vert pour l'état final
- Rouge pour les murs
- **a, b, c, d** pour les actions
- Un système de **récompense**

Un problème intermédiaire pour se lancer

Définitions



Le Q-learing :

- Une fonction de valuation :

$$V^{\pi}(s,a) \qquad (1)$$

- Pour se déplacer :

$$s' = \max_{a}(V^{\pi}(s,a)) \qquad (2)$$

- La récompense :

$$R(s,a) \qquad (3)$$

Algorithme et équation de Bellman

Initialisation

On définit les $V^{\pi}(s,a)$ aléatoirement

Récurrence

- On effectue une simulation grâce à la formule (2)
- Sur chaque état alors visité, on applique l'équation de Bellman :

$$V^{\pi}_{t+1}(s,a) = R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^{\pi}_{t}(s') \tag{4}$$

$$\Leftrightarrow {}^{6} V^{\pi}_{t+1}(s,a) = R(s,a) + \gamma V^{\pi}_{t}(s') \tag{5}$$

. Ici l'équivalence vient du fait que l'environnement est déterministe

Terminaison

On arrête l'algorithme une solution optimal est trouvée ou si une limite de temps est dépassée

Performance de la méthode

ici, il faut une image des performance au cours du temps sur le Q learning, je n'en ai pas trouver

Limite de la méthode

Physique

- Temps d'exécution
- Faible adaptivité
- Difficulté malgrès l'environnement simple

Limite de la méthode

Physique

- Temps d'exécution
- Faible adaptivité
- Difficulté malgrès l'environnement simple

Amélioration

- Un environnement plus réaliste
- Une meilleur adaptivité
- Une vitesse de calcul plus importante

Sommaire

Objectifs et contraintes

Objectif

- Prise en compte de l'inertie
- Prise en compte des frottements visqueux
- Prise en compte des caractéristiques physiques du bateau
- Un environnement qui représente un port

Objectifs et contraintes

Objectif

- Prise en compte de l'inertie
- Prise en compte des frottements visqueux
- Prise en compte des caractéristiques physiques du bateau
- Un environnement qui représente un port

Contrainte

- Le modèle doit être très rapide d'exécution
- Autoriser l'exécution en parallèle
- Être représentable visuellement

Visuellement

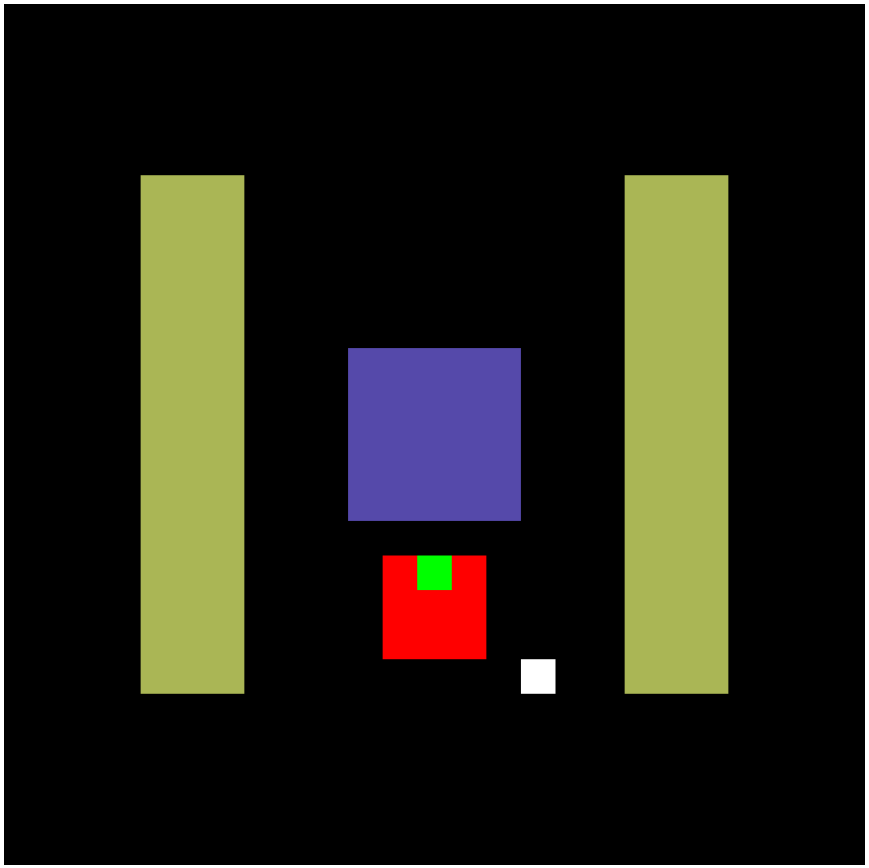
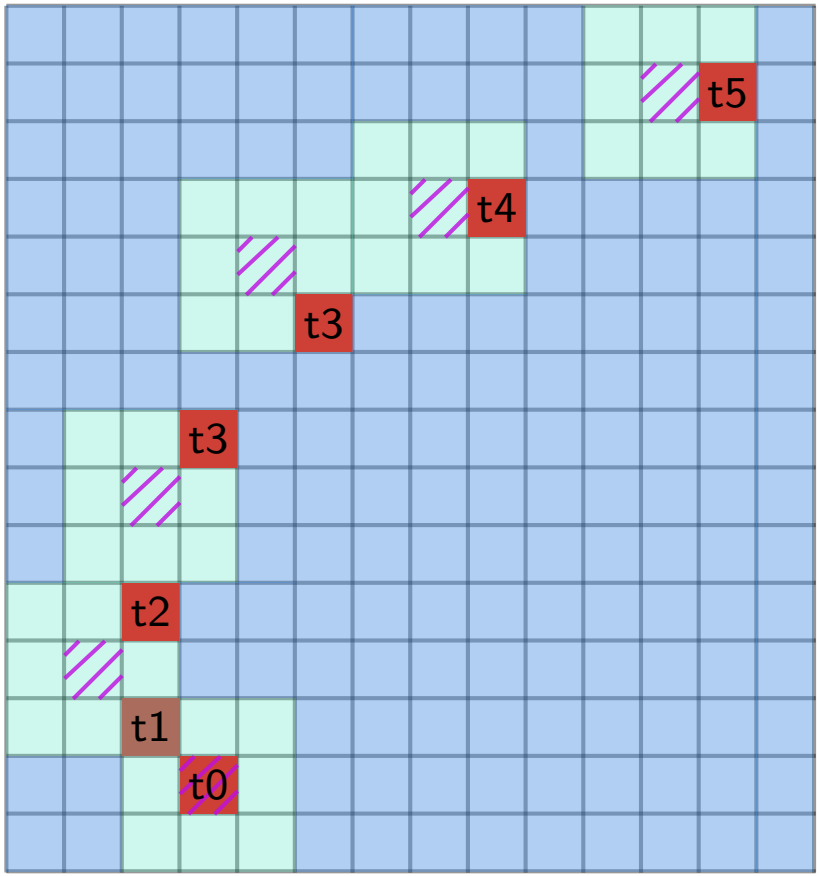


FIGURE – Rendu visuel de notre environnement [7]

- Un environnement **discrétisé**
- Blanc pour le bateau
- Beige pour les murs
- Violet pour l'objectif
- Les cases rouges et vertes **montre les actions**

7. Les actions sont **seulement affichées pour l'utilisateur**, elles ne font pas parties de l'environnement

Visuellement



- En rouge les positions successives du bateau
- En violet, prise en compte de l'inertie (répétition du déplacement)
- En vert les choix d'actions successifs (rayon 1 de autour de violet [8])
- Sous python, **Numpy** permet la **vectorisation** et donc les **parties simultanées** (1000 parties prennent le même temps que une ou deux parties)

FIGURE – Exemple de trajectoire de bateau

8. Ici, la zone est carré mais la forme peut varier pour augmenter l'aspect réaliste du modèle et s'adapter aux caractéristiques même du bateau.

Sommaire

Définissions et notations

La Politique et ses fonctions Gain, Q-Value, Value et Reward associées

- La Politique

$$\pi_\theta(s) = (p_i)_{i \in [\![1,ac]\!]} / \sum_{i=1}^{ac} p_i = 1 \tag{6}$$

- Le Gain

$$G_t = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{7}$$

- La Q-value

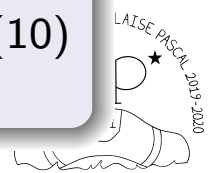
$$Q^\pi(s,a) = \mathbb{E}_{a\sim\pi}[G_t | S_t = s, A_t = a] \tag{8}$$

- La Value

$$V^\pi(s) = \mathbb{E}_{a\sim\pi}[G_t | S_t = s] \tag{9}$$

- La Recompense, fonction Reward

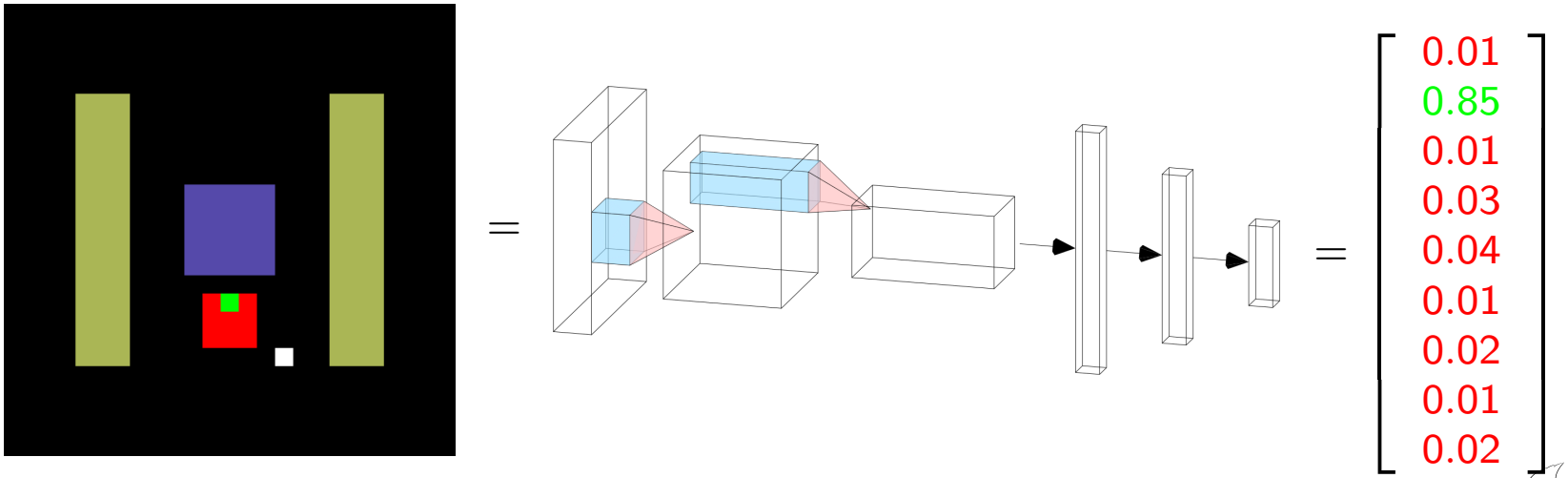
$$J(\theta) = \sum_s d^\pi(s) V^\pi(s) = \sum_s d^\pi(s) \sum_a \pi_\theta(a|s) Q^\pi(s,a) = \mathbb{E}_\pi[Q^\pi(s,a)] \tag{10}$$



La Politique

Un réseau de neurones de convolution maison [9] pour $\pi_\theta(s)$

- s correspond à l'entrée
- $\pi_\theta(s)$ correspond à la sortie
- θ correspond aux poids et biais du réseau



9. Nous utilisons ici notre implémentation sans librairie spécialisée du Convolutional Neural Network

L'entrainement

L'initialisation

On définit une structure pour le réseau de neurones où les poids et biais sont définis aléatoirement

Par récurrence (En époques) [10]

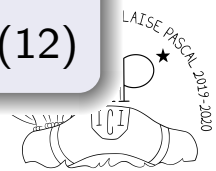
- On effectue P parties en parallèle, on récupère :

$$\begin{cases} S_0^1, A_0^1, R_0^1, \cdots, S_{f_1-1}^1, A_{f_1-1}^1, R_{f_1-1}^1, S_{f_1}^1 \\ \vdots \\ S_0^P, A_0^P, R_0^P, \cdots, S_{f_P-1}^P, A_{f_P-1}^P, R_{f_P-1}^P, S_{f_P}^P \end{cases}$$

- Pour tout $i \in [\![1, P]\!]$ et $t \in [\![0, f_i - 1]\!]$

$$G_t^i = \sum_{k=O}^{f_i-t-2} \gamma^k R_{t+k+1}^i \tag{11}$$

$$\theta \leftarrow \theta + \alpha\gamma^t G_t^i \nabla_\theta \ln \pi_\theta(A_t^i | S_t^i) \tag{12}$$



10. Ceci résulte d'un théorème majeur sur le Policy Gradients

Quelques précisions sur les gradients : La rétropropagation

Initialisation

On calcul les gradients de la dernière couche du réseau

Par récurrence

On rétro propage les gradients sur l'ensemble du réseau de neurones grâce à l'astuce :

$$\frac{\partial \pi_\theta(s)}{\partial \theta} = \frac{\partial \pi_\theta(s)}{\partial a} \cdot \frac{\partial a}{\partial \theta}$$

Ce qui nous permet en accumulant ce principe de remonter couches par couches le réseau de neurones.

Une implémentation naïve de Policy Gradient

Avec un DNN

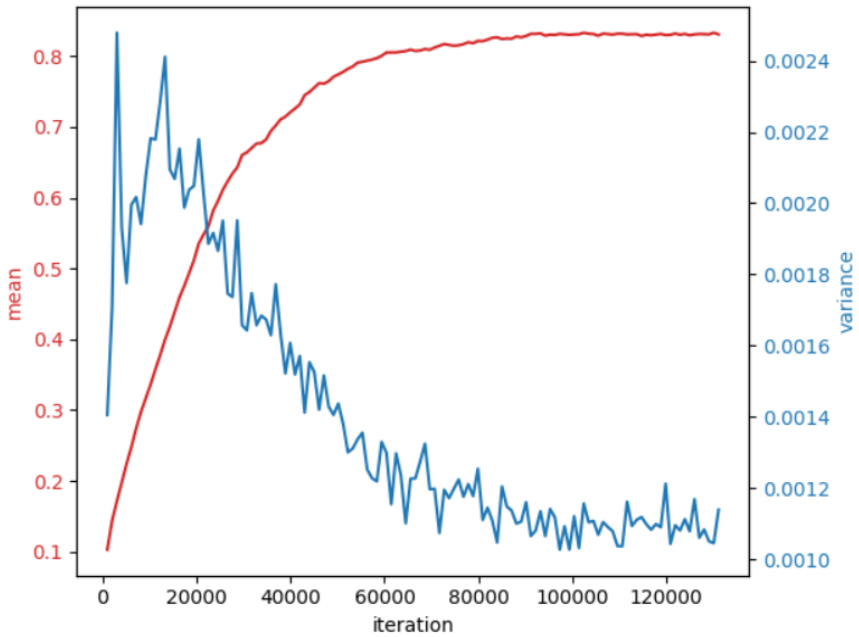


FIGURE – 135000 époques de 100 parties, 82.35% de réussite en 5h31m46s sur un cœur de CPU à 3,7 GHz

- En rouge les performances moyennes
- En bleu la variance moyenne

Une implémentation naïve de Policy Gradient

Avec un CNN

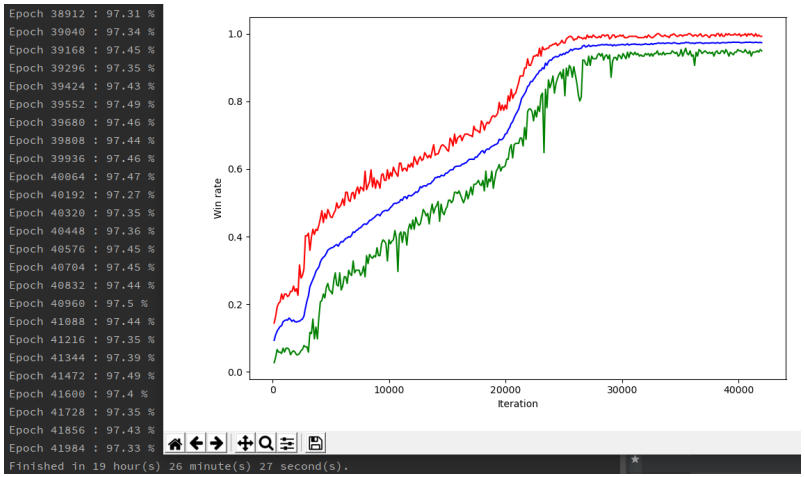


FIGURE – 42000 époques de 200 parties, 97.49% de réussite en 19h26m27s sur 1 cœur de CPU à 3.7GHz

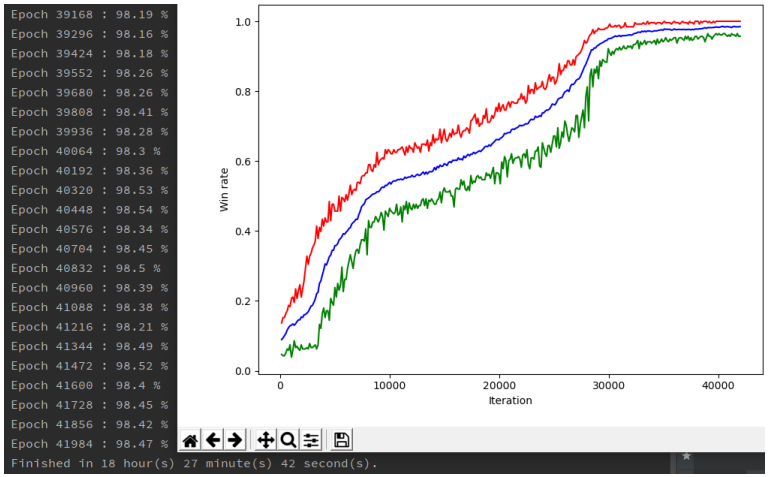


FIGURE – 42000 époques de 200 parties, 98.54% de réussite en 18h27m42s sur 1 cœur de CPU à 3.7GHz

Une implémentation plus juste du Policy Gradient

Avec un CNN

Ici mettre les futures courbes

Objectif

- Réussir à faire stationner un bateau dans un port
- Programmer notre propre algorithme d'apprentissage par renforcement sans utiliser de librairie dédiée. ✓
- Créer une simulation discrète et réaliste d'un déplacement de bateau prenant en compte l'inertie et la viscosité. ✓
- Comprendre et réussir à manipuler les concepts sur lesquels sont basés l'intelligence artificielle. ✓
- Implémenter différentes technologies pour pouvoir comparer les performances et trouver la meilleure solution technique à notre problème. ∼

Ouverture

- Implémentation de l'Actor Critic
- Meilleurs algorithmes
- Simulation encore plus réaliste

Un fait maison

Il faudra peut être détailler ici le CNN et DNN et qu'est ce que le fait main"

Retour

Développement de $\nabla_\theta V^\pi(s)$

$$
\begin{aligned}
\nabla_\theta V^\pi(s) &= \nabla_\theta \left[\sum_a \pi_\theta(a|s) Q^\pi(s,a)\right] \\
&= \sum_a \left[\nabla_\theta \pi_\theta(a|s) Q^\pi(s,a) + \pi_\theta(a|s) \nabla_\theta Q^\pi(s,a)\right] \\
&= \sum_a \left[\nabla_\theta \pi_\theta(a|s) Q^\pi(s,a) + \pi_\theta(a|s) \nabla_\theta \sum_{s',r} p(s',r|s,a)(r + V^\pi(s'))\right] \\
&= \sum_a \left[\nabla_\theta \pi_\theta(a|s) Q^\pi(s,a) + \pi_\theta(a|s) \nabla_\theta \sum_{s'} p(s'|s,a) \nabla_\theta V^\pi(s'))\right] \\
&= \Phi(s) + \sum_a \left[\pi_\theta(a|s) \nabla_\theta \sum_{s'} p(s'|s,a) \nabla_\theta V^\pi(s'))\right] \\
&= \Phi(s) + \sum_{s'} \sum_a \left[\pi_\theta(a|s) \nabla_\theta p(s'|s,a) \nabla_\theta V^\pi(s'))\right] \\
&= \Phi(s) + \sum_{s'} \rho^\pi(s \to s', 1) \nabla_\theta V^\pi(s')) \\
&= \Phi(s) + \sum_{s'} \rho^\pi(s \to s', 1) \nabla_\theta \left[\sum_a \pi_\theta(a|s) Q^\pi(s',a)\right] \\
&= \Phi(s) + \sum_{s'} \rho^\pi(s \to s', 1) \Phi(s') + \sum_{s''} \rho^\pi(s \to s'', 1) \nabla_\theta V^\pi(s'')) \\
&= \cdots \\
&= \sum_{\tilde{s}} \sum_k \rho^\pi(s \to \tilde{s}, k) \phi(\tilde{s})
\end{aligned}
$$

Développement de $\nabla_\theta J(\theta)$

Ici, nous devons supposer que les parties sont finies (i.e. $k \in [\![0, f_i - 1]\!]$ au lieu de $k \in [\![0, +\infty[\![$)

$$
\begin{aligned}
\nabla_\theta J(\theta) &= \nabla_\theta \sum_{s_0} d^\pi(s_0) V^\pi(s_0) \\
&= \nabla_\theta \sum_{s_0} p(s_0) V^\pi(s_0) \\
&= \sum_{s_0} p(s_0) \nabla_\theta V^\pi(s_0) \\
&= \sum_{s_0} p(s_0) \sum_s \sum_{k=0}^{f_i - 1} \rho^\pi(s_0 \to s, k) \phi(s) \\
&= \sum_{s_0} p(s_0) \left[\sum_{k=0}^{f_i - 1} \sum_s \rho^\pi(s_0 \to s, k) \right] \sum_s \frac{\sum_k^{f_i} \rho^\pi(s_0 \to s, k)}{\sum_s \sum_{k=0}^{f_i - 1} \rho^\pi(s_0 \to s, k)} \phi(s) \\
&= \sum_{s_0} p(s_0) \left[\sum_{k=0}^{f_i - 1} 1 \right] \sum_s d^\pi(s) \phi(s) \\
&= \sum_{s_0} p(s_0) f_i \sum_s d^\pi(s) \phi(s) \\
&= \sum_{s_0} p(s_0) f_i \sum_s d^\pi(s) \sum_a \nabla_\theta \pi_\theta(a|s) Q^\pi(s, a) \\
&= f_i \left[\sum_{s_0} p(s_0) \right] \left[\sum_s d^\pi(s) \sum_a \nabla_\theta \pi_\theta(a|s) Q^\pi(s, a) \right] \\
&= f_i \left[\sum_s d^\pi(s) \sum_a \nabla_\theta \pi_\theta(a|s) Q^\pi(s, a) \right]
\end{aligned}
$$

Utilisation de $\nabla_\theta J(\theta)$

Si l'on suppose maintenant que tout les parties ont une durée proche :

$$\begin{aligned}
\nabla_\theta J(\theta) &= f_i \left[\sum_s d^\pi(s) \sum_a \nabla_\theta \pi_\theta(a|s) Q^\pi(s,a)\right] \\
&= \propto \left[\sum_s d^\pi(s) \sum_a \nabla_\theta \pi_\theta(a|s) Q^\pi(s,a)\right] \\
&= \propto \left[\sum_s d^\pi(s) \sum_a \pi_\theta(a|s) \frac{\nabla_\theta \pi_\theta(a|s)}{\pi_\theta(a|s)} Q^\pi(s,a)\right] \\
&= \propto \left[\sum_s d^\pi(s) \sum_a \pi_\theta(a|s) \nabla_\theta Q^\pi(s,a) \ln \pi_\theta(a|s)\right] \\
&= \mathbb{E}_{s\sim d^\pi, a\sim\pi_\theta}[Q^\pi(s,a) \nabla_\theta \ln \pi_\theta(a|s)]
\end{aligned}$$

Il faut maintenant revenir à la machine qui nous permettra d'obtenir A_t^i, S_t^i, R_t^i :

$$\begin{aligned}
\nabla_\theta J(\theta) &= \mathbb{E}_{s\sim d^\pi, a\sim\pi_\theta}[Q^\pi(s,a) \nabla_\theta \ln \pi_\theta(a|s)] \\
&= \mathbb{E}_{s\sim d^\pi, a\sim\pi_\theta}[G_t^i \nabla_\theta \ln \pi_\theta(A_t^i|S_t^i)]
\end{aligned}$$

On cherche à augmenter $J(\theta)$ d'où la monté de gradient, d'où des modifications sur les paramètres selon le gradients, il ne faut pas oublier de recoefficienter le tout en fonction de la temporalité avec γ^t :

$$\theta \leftarrow \theta + \alpha \gamma^t G_t^i \nabla_\theta \ln \pi_\theta(A_t^i|S_t^i)$$

Retour