I will try to address the reinforcement learning question of solving a Reacher task for an underwater robotic arm, in which an agent learns to reach arbitrary target positions with actuations of a robotic arm by using trial and error.

Based on what I have read in some papers, there are already several proven control algorithms that can solve this kind of robot arm Reacher task easily in a simulation but not underwater.

So, difficulties of this task can be from the part which we pull it out of the simulation and put it into the real-world and also from the fact that our robot arm is underwater.

We describe the steps and elements of setting up real-world RL tasks including the medium of data transmission, concurrency, ordering and delays of computations, low-level actuation types, and frequencies of operating them.

And for now, maybe I will start with constructing an underwater simulation I guess, and implementing some existed RL model.

Then I would like to introduce some information on the reinforcement learning.

Learning from interaction is a foundational idea underlying nearly all theories of learning and intelligence. And the reinforcement learning a computational approach to learning from interaction.

Reinforcement learning is learning what to do—how to map situations to actions—so as to maximize a numerical reward signal. The learner is not told which actions to take, but instead must discover which actions yield the most reward by trying them.

In the most interesting and challenging cases, actions may affect not only the immediate reward but also the next situation and, through that, all subsequent rewards. These two characteristics—trial-and-error search and delayed reward—are the two most important distinguishing features of reinforcement learning.

One can identify three main sub elements of a reinforcement learning system: a policy, a reward signal, and a value function, in which the reward signal indicates what is good in an immediate sense, a value

function specifies what is good in the long run. The agent’s sole objective is to maximize the total reward it receives over the long run.

In fact, the most important component of almost all reinforcement learning algorithms we consider is a method for efficiently estimating values.

For the learning method, we use the rllab implementation of TRPO (Schulman et al. 2015, Duan et al. 2016), a popular learning method with robust performance with respect to its hyper-parameters.

Je vais d'aborder la question de l'apprentissage par renforcement pour résoudre une tâche Reacher pour un bras robotique sous-marin, dans laquelle un agent apprend à atteindre des positions cibles arbitraires avec des actions d'un bras robotique en utilisant l'essai et l'erreur.

D'après ce que j'ai lu dans certains articles, il existe déjà plusieurs algorithmes de contrôle éprouvés qui peuvent résoudre ce type de tâche pour un bras robotique facilement dans une simulation, et pas sous l'eau.

Du coup, les difficultés de cette tâche peuvent provenir de la partie où on la sort de la simulation et la met dans le monde réel et aussi du fait que notre bras robotique est sous l'eau.

D’après un article, les étapes et les éléments de la mise en place de tâches de reinforcement learning dans le monde réel, y compris le moyen de transmission des données, la simultanéité, la commande et les délais des calculs, les types d'actionnement de bas niveau et les fréquences de leur fonctionnement.

Et pour l'instant, je vais peut-être commencer par construire une simulation sous-marine, je suppose, et mettre en œuvre un modèle RL qui existe déjà.

Du coup là j’aimerais introduire quelques informations sur l'apprentissage par renforcement.

L'apprentissage par interaction est une idée fondamentale qui se cache derrière presque toutes les théories de l'apprentissage et de l'intelligence. L'apprentissage par renforcement est une approche computationnelle de l'apprentissage par interaction.

L'apprentissage par renforcement consiste à apprendre ce qu'il faut faire - comment associer des situations à des actions - de manière à maximiser un signal de récompense numérique. On ne dit pas à l'apprenant quelles actions il doit effectuer, mais il doit découvrir les actions les plus rentables en les essayant.

Dans les cas les plus intéressants et les plus difficiles, les actions peuvent affecter non seulement la récompense immédiate, mais aussi la situation suivante et, par conséquent, toutes les récompenses qui suivront. Ces deux caractéristiques - recherche par essais et erreurs et récompense retardée - sont les deux plus importantes caractéristiques distinctives de l'apprentissage par renforcement.

On peut identifier trois sous-éléments principaux d'un système d'apprentissage par renforcement : une stratégie, un signal de récompense et une fonction de valeur, dans lesquels le signal de récompense indique ce qui est bon dans un sens immédiat, une fonction de valeur spécifie ce qui est bon à long terme. Le seul objectif de l'agent est de maximiser la récompense totale qu'il reçoit sur le long terme.

En fait, le composant le plus important de presque tous les algorithmes d'apprentissage par renforcement que on considère est une méthode d'estimation efficace des valeurs.