Contrôle d'un Bras Robotique via l'Apprentissage par Renforcement

Rapport de Stage de 2ème année

30 septembre 2024

Romain Bornier
fise 2025
Spécialité Robotique
romain.bornier@ensta-bretagne.fr

Résumé

Ce projet s'inscrit dans le cadre d'un stage international de quatre mois au laboratoire du Professeur Yoshida à l'Université des Sciences de Tokyo, sous la supervision du Dr. Marwan Hamze. L'objectif principal était de contribuer à la recherche sur l'application de l'apprentissage par renforcement pour le contrôle de bras robotiques.

Mon travail a consisté à développer et à implémenter des algorithmes de contrôle pour un bras robotique, tant en simulation qu'en environnement réel. L'apprentissage par renforcement permet d'éviter les modèles cinématiques complexes, offrant ainsi au robot la capacité d'optimiser son comportement par interaction directe avec son environnement.

J'ai axé mes efforts sur l'optimisation du contrôle du bras robotique xArm6, en adaptant des méthodes issues de la littérature scientifique. J'ai d'abord testé ces algorithmes en simulation avant de les appliquer dans des environnements réels pour évaluer leur robustesse. Mon but était d'acquérir des compétences en apprentissage par renforcement appliqué au contrôle de robots humanoïdes, en vue de contrôler le robot Kaleido de Kawasaki, mesurant 1,80 m et pesant 80 kg.

Ce projet m'a permis de renforcer mes compétences techniques en robotique et en intelligence artificielle tout en contribuant à la recherche appliquée dans ce domaine en pleine expansion.

Abstract

This project is part of a four-month international internship at Professor Yoshida's laboratory at the Tokyo University of Science, under the supervision of Dr. Marwan Hamze. The main objective was to contribute to research on the application of reinforcement learning for the control of robotic arms.

My work involved developing and implementing control algorithms for a robotic arm, both in simulation and in real-world environments. Reinforcement learning allows for the avoidance of complex kinematic models, enabling the robot to optimize its behavior through direct interaction with its environment.

I focused on optimizing the control of the xArm6 robotic arm by adapting methods from the scientific literature. I initially tested these algorithms in simulation before applying them in real-world environments to assess their robustness. My goal was to acquire skills in reinforcement learning applied to the control of humanoid robots, with the aim of controlling the Kawasaki Kaleido robot, which measures 1.80 m and weighs 80 kg.

This project enabled me to enhance my technical skills in robotics and artificial intelligence while contributing to applied research in this rapidly growing field.

Remerciements

Je tiens à exprimer ma profonde gratitude à l'ensemble des membres du Laboratoire Yoshida pour leur accueil chaleureux, leur aide précieuse et leur volonté de me faire découvrir la culture japonaise. Je souhaite également adresser mes remerciements particuliers au Dr Marwan Hamze, dont l'accompagnement et les conseils m'ont été précieux et m'ont permis de développer mon intérêt pour la recherche scientifique.

Table des matières

Chapitre I

Présentation du Laboratoire

Sommaire

I.1 Présentation du Laboratoire

Mon stage s'est déroulé à l'Université des Sciences de Tokyo, à Katsushika, au sein du laboratoire Yoshida, également appelé Laboratoire de Robotique Interactive. Ce laboratoire, fondé en 2022, vise à développer des robots plus intelligents en s'inspirant du fonctionnement du corps humain. L'objectif est de concevoir des robots capables de reproduire des mouvements humains ou de coopérer avec les humains dans diverses tâches.

Les recherches portent notamment sur le contrôle de robots humanoïdes, basé soit sur des modèles mathématiques précis, soit sur des méthodes d'intelligence artificielle. Des travaux sont également réalisés sur les capteurs de force et les dispositifs de mesure pour améliorer la précision des mouvements robotiques. Par ailleurs, certains projets explorent l'augmentation des capacités humaines à l'aide de systèmes robotiques.

Bien que le laboratoire dispose de ressources pour la construction de robots, de nombreux projets sont réalisés sur des plateformes commerciales, telles que le robot humanoïde Kaleido de 1,80 m et 80 kg, développé par Kawasaki, ou encore le bras robotique à six axes de UFactory. Le laboratoire utilise également des imprimantes 3D pour le prototypage de composants.

I.2 Présentation de l'équipe

Le laboratoire Yoshida compte une dizaine de membres, dont cinq étudiants en Bachelor et Master, encadrés soit par Dr Yoshida, soit par le Dr Sasaki. En plus de leurs études, les étudiants participent à des projets de recherche. L'entraide entre les membres est une caractéristique marquante de l'équipe.

Chaque mercredi après-midi, une réunion est organisée pour que chacun présente, une fois par mois, l'avancement de ses travaux. Ces rencontres regroupant post-doctorants et étudiants permettent de renforcer l'esprit d'équipe.

Le laboratoire est encore modeste au sein de l'université, comme l'ont remarqué d'autres stagiaires de l'ENSTA Bretagne, mais il aspire à grandir en recrutant davantage d'étudiants japonais et internationaux, y compris des doctorants et stagiaires. Durant mon stage, j'ai travaillé aux côtés de personnes de nationalités diverses, ce qui a facilité mon intégration. Les francophones parlant également japonais ont facilité les échanges avec les autres membres du laboratoire et m'ont aidé à mieux comprendre la culture japonaise.

L'ambiance collaborative a favorisé de nombreux échanges informels, comme des sorties au restaurant et une soirée takoyaki, renforçant les liens entre les membres.

Mon travail a été supervisé par le Dr Marwan Hamze, un expert en contrôle des robots humanoïdes par modèle physique, qui est en formation pour devenir spécialiste en apprentissage par renforcement. Il est actuellement impliqué dans des recherches sur le robot humanoïde Kaleido de Kawasaki, où il explore l'exploitation du multicontact grâce à l'intelligence artificielle.

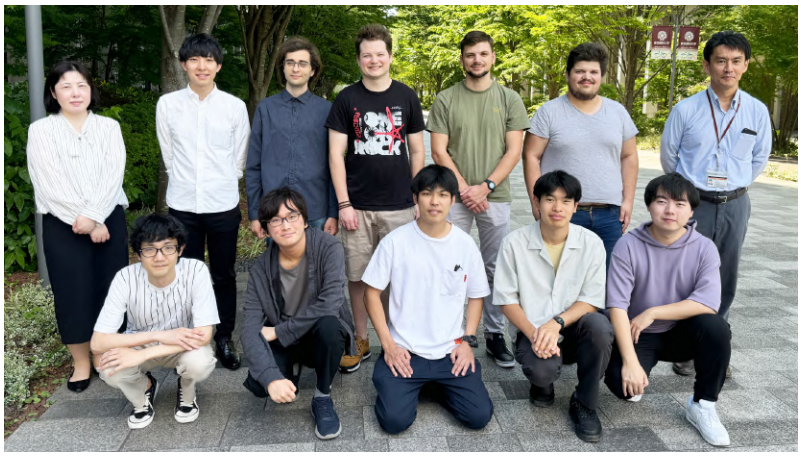

FIGURE I.1 – L'équipe du laboratoire Yoshida

I.3 La place du laboratoire dans la recherche

Le laboratoire Yoshida se tourne de plus en plus vers des collaborations internationales. Durant mon stage, j'ai assisté à plusieurs conférences présentées par des chercheurs d'Allemagne, de France et du Canada. Ces événements étaient suivis de présentations où les étudiants exposaient leurs projets, ce qui m'a permis d'échanger avec ces chercheurs et de mieux saisir les avancées en robotique humanoïde et collaborative. Une conférence notable portait sur l'impact écologique de la robotique, donnée par un chercheur français.

Le laboratoire a également participé à des salons majeurs, tels que l'ICRA (International Conference on Robotics and Automation), où il a pu présenter plusieurs projets.

La présence de nombreux francophones s'explique par la collaboration avec le CNRS-AIST Joint Robotics Lab (JRL). Ce laboratoire commun, entre le CNRS en France et l'AIST au Japon, est situé à Tsukuba et se concentre sur l'augmentation de l'autonomie fonctionnelle des robots, en particulier des humanoïdes. De nombreux chercheurs circulent entre ces deux laboratoires, notamment sur les questions liées à l'apprentissage par renforcement.

Ces échanges m'ont permis de me sentir pleinement intégré dans un groupe de recherche international, au-delà du simple rôle de stagiaire.

Chapitre II

Dynamique du Stage : Défis, Changements et Aspects Humains

Sommaire

II.1 Enjeux et définition initiale du stage

En général, pour contrôler un robot, la méthode la plus répandue consiste à étudier son modèle dynamique et son interaction avec l'environnement.

Ces contrôleurs basés sur la modélisation reposent sur la conception d'un modèle mathématique détaillé de la dynamique du robot et de l'environnement pour guider la synthèse du contrôle. Le pendule inversé est un modèle simplifié [Kea01] qui s'est montré très utile pour la locomotion des robots bipèdes [Kea10].

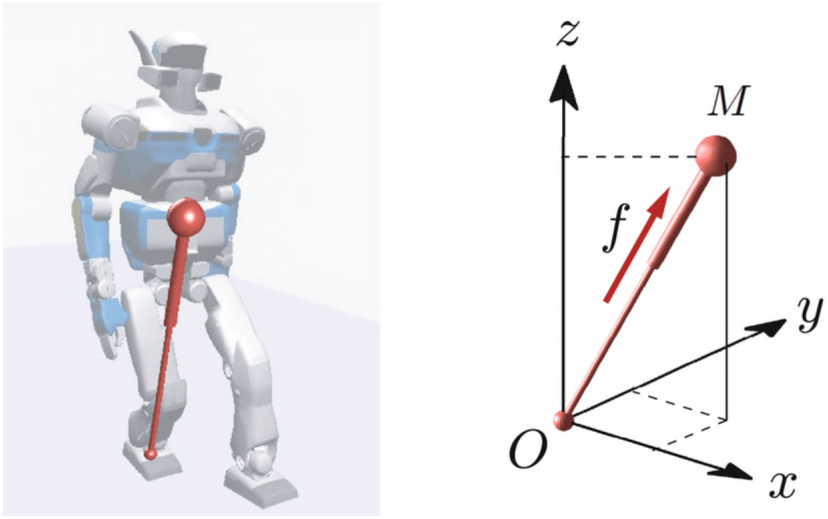

FIGURE II.1 – Illustration du modèle du pendule inversé [Kea05]

Malgré leur efficacité, ces contrôleurs nécessitent des calculs détaillés et peuvent ne pas représenter avec précision la physique réelle du robot ou de l'environnement à cause des erreurs inhérentes aux modèles.

D'autre part, les contrôleurs basés sur l'intelligence artificielle, en particulier l'apprentissage par renforcement, se concentrent sur l'entraînement des robots par essais et erreurs dans un environnement simulé ou réel. Ils apprennent des comportements optimaux en recevant des récompenses ou des pénalités selon leurs actions, améliorant progressivement leurs performances sans nécessiter une connaissance explicite de la dynamique du robot. En comprenant le fonctionnement interne du robot et sa réponse aux diverses entrées, ces contrôleurs peuvent prédire les états futurs et ajuster les actions pour atteindre les résultats souhaités avec plus de précision.

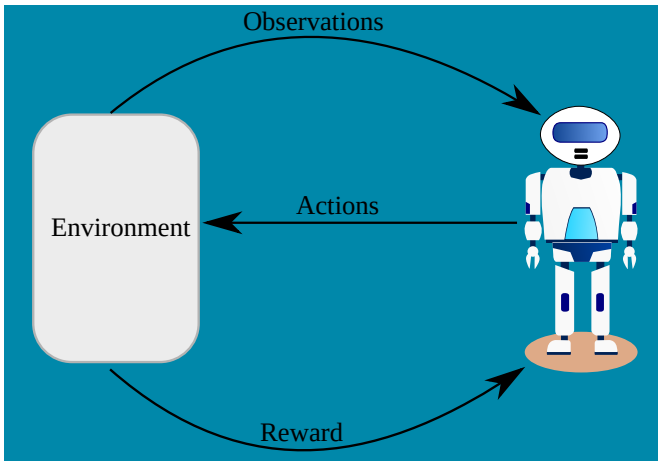

FIGURE II.2 – Illustration de l'apprentissage par renforcement pour un robot.

L'enjeu principal de mon stage était de contribuer aux travaux de recherche visant à appliquer la technique de contrôle par apprentissage par renforcement au robot humanoïde Kaleido du laboratoire. Ce robot, d'une taille de 1m80 et d'un poids de 80 kg, est équipé d'une trentaine d'actionneurs afin de simuler au mieux les mouvements humains. Les recherches autour de ce projet sont menées par une équipe de cinq chercheurs, dont un expert en intégration de l'intelligence artificielle, ainsi que le Dr Marwan Hamze, le post-doctorant qui supervisait mon travail.

La complexité de ce projet réside dans le nombre de capteurs et d'actionneurs du robot, ce qui rend l'apprentissage par renforcement plus difficile et les tests réels potentiellement dangereux. En effet, avec cette approche, on ne contrôle pas directement les actionneurs nécessaires pour accomplir une tâche spécifique. Le robot décide lui-même quels actionneurs activer. Cependant, si l'entraînement n'est pas correctement effectué, le robot pourrait activer des actionneurs de manière aléatoire, augmentant ainsi les risques pour les opérateurs. Par exemple, lors d'une tâche de simple assise, il pourrait activer brusquement les actionneurs des bras, ce qui serait dangereux, car un bras de ce robot peut peser jusqu'à 15 kg.

Pour prévenir ces risques, plusieurs algorithmes de sécurité doivent être développés. Initialement, mon rôle dans ce projet était d'assister le Dr Marwan dans ses recherches et de contribuer aux travaux de l'équipe sur ce sujet complexe.

FIGURE II.3 – Robot humanoïde Kaleido, société Kawasaki.

II.2 Changement de direction

Cependant, ce rôle exigeait des compétences solides en robotique humanoïde et, surtout, en intelligence artificielle, domaines que je ne maîtrisais pas au début de mon stage. De plus, étant donné que mon tuteur était lui-même en formation en intelligence artificielle, cela limitait les tâches intéressantes qu'il pouvait me confier, restreignant ainsi mon implication dans ses projets.

Dans un premier temps, il m'a tout de même demandé de me former à l'IA en m'appuyant sur divers documents qu'il me fournissait, et d'étudier les recherches précédemment réalisées, notamment celles appliquées à un modèle de simulation 3D du robot Kaleido. Cependant, je me suis rapidement retrouvé dans une impasse, n'ayant pas de projet suffisamment structuré pour occuper plusieurs mois.

Face à ce constat, le Dr Hamze et moi avons réfléchi à une solution pour que mon travail devienne à la fois utile et stimulant. C'est alors que nous avons décidé de nous tourner vers le bras robotique xArm6, bras possédant 6 actionneurs, disponible au laboratoire mais jusque-là peu exploité.

FIGURE II.4 – Bras robotique xArm6, société Ufactory.

L'une des principales difficultés du contrôle du robot Kaleido est liée à son grand nombre d'actionneurs, ce qui complique l'apprentissage par renforcement. En revanche, travailler sur un robot au fonctionnement similaire à un humanoïde, mais avec un nombre réduit de capteurs, était une solution idéale. Cela me permettait de me familiariser progressivement avec cette nouvelle approche, tout en contribuant de manière plus concrète aux recherches du Dr Hamze. Grâce à ce robot, je pouvais tester rapidement divers algorithmes et tâches, en lien direct avec le projet initial.

En effet, alors qu'il faut plus de 24 heures pour entraîner un robot de la taille de Kaleido, le temps de calcul pour le bras robotique sur lequel je travaillais n'était que d'environ une heure. De plus, ce projet m'a permis de développer une plateforme simple pour aborder les concepts d'apprentissage par renforcement. J'ai notamment créé un dépôt GitHub détaillant les démarches et les lignes de commande essentielles. Le laboratoire, souhaitant accroître ses compétences en intelligence artificielle, pourra utiliser mon travail comme guide pour les nouveaux arrivants intéressés par ce sujet.

Avec l'accord de mon directeur de laboratoire, mon sujet de stage est ainsi devenu : "Contrôle d'un bras robotique à l'aide de l'apprentissage par renforcement : développement d'une simulation, de tests, et d'un guide explicatif".

Ce nouveau sujet était beaucoup mieux adapté à mon profil, me permettant de découvrir et d'approfondir mes connaissances en IA et en intégration de l'IA dans des simulations ainsi que dans des systèmes robotiques réels. De plus, il m'a offert une grande autonomie, me poussant constamment à apprendre et à améliorer mes expérimentations.

II.3 Les enjeux humains

Au-delà des défis techniques, l'un des plus grands enjeux de mon stage a été l'intégration au sein d'un environnement culturellement très différent, et principalement composé de collègues japonais. Bien que certains membres du laboratoire soient francophones, la majorité des échanges se faisaient en japonais, ce qui rendait l'intégration d'autant plus complexe, d'autant plus que la plupart de mes collègues maîtrisaient l'anglais de manière limitée. Cette barrière linguistique a donc constitué un obstacle important, tant pour la communication au travail que pour la vie quotidienne.

Avant mon départ, j'avais anticipé ces difficultés en prenant des cours de japonais basiques, pensant que cela me permettrait de gérer des situations courantes en dehors de Tokyo, où je croyais que l'anglais serait plus répandu. Toutefois, à mon arrivée, j'ai rapidemment réalisé que l'anglais était beaucoup moins utilisé que je ne l'avais imaginé, même dans les grandes villes. Cela m'a poussé à intensifier mes efforts pour apprendre le japonais, mais l'apprentissage de cette langue, très éloignée des langues latines, s'est révélé particulièrement ardu, surtout dans un contexte où les ressources pour étrangers n'étaient pas toujours accessibles.

J'ai cherché des cours de japonais au sein de l'université pour m'améliorer, mais étant donné que le campus était relativement récent et accueillait peu d'étudiants internationaux, ces cours n'étaient pas encore disponibles. Mon directeur de laboratoire, conscient de cette difficulté, a fait remonter cette problématique à la direction de l'établissement pour améliorer l'accueil des futurs stagiaires et chercheurs internationaux. Ce soutien de sa part a montré l'importance de l'adaptation culturelle dans ce type de collaboration scientifique.

Malgré ces défis linguistiques, cette barrière s'est révélée être une opportunité d'enrichissement personnel. Avec mes collègues japonais, nous avons échangé, non sans difficultés, sur

nos cultures respectives. Ces échanges étaient souvent lents et compliqués par les différences de langue, mais ils ont permis de renforcer nos liens, de mieux comprendre nos modes de vie et de travail, et d'élargir nos perspectives culturelles. Ces moments de partage ont été précieux, car ils ont permis de dépasser les simples interactions professionnelles pour construire des relations plus profondes et humaines.

Cependant, les différences culturelles et linguistiques étaient encore plus marquées dans mes interactions avec l'administration, que ce soit à l'université, à la mairie, ou dans les démarches liées à mon logement. La moindre formalité administrative devenait un véritable défi, car le personnel ne parlait souvent que japonais, ce qui rendait les procédures longues et parfois déroutantes. J'ai dû faire preuve de patience et d'adaptabilité pour surmonter ces difficultés.

Cette expérience de quatre mois au Japon m'a permis de comprendre qu'il est tout à fait possible de vivre dans un pays dont la culture et la langue sont complètement différentes des notre. J'ai développé une plus grande flexibilité face aux différences culturelles et j'ai appris à m'adapter rapidement à un nouvel environnement. Ce stage m'a également montré l'importance de l'ouverture d'esprit, de la curiosité et de la résilience face aux défis interculturels, des qualités essentielles dans un contexte professionnel international.

Chapitre III

Compte-Rendu des Activités et Résultats du Stage

Sommaire

III.1 Apprentissage du Reinforcement Learning

III.1.1 Formation Initiale en Intelligence Artificielle

Sans connaissance préalable en intelligence artificielle et en apprentissage par renforcement, j'ai entrepris une formation autonome pour acquérir les bases nécessaires à mon stage. J'ai commencé par suivre des vidéos éducatives de Thibault Neveu, ancien membre du comité de recherche en IA de Samsung. Ces vidéos m'ont permis de me familiariser avec des concepts fondamentaux tels que le deep learning, les réseaux de neurones et les bases de l'apprentissage par renforcement. Cependant, il me manquait encore une compréhension

plus appliquée et orientée vers la robotique, ce qui est essentiel pour les travaux que je devais mener.

Pour approfondir ces connaissances, j'ai sollicité les conseils de mon professeur, M. Benoît Zerr, qui m'a recommandé la lecture du livre *Reinforcement Learning: An Introduction* de Richard S. Sutton et Andrew G. Barto [SB15]. Par la suite, il m'a mis en relation avec M. Gilles Le Chenadec, professeur de machine learning à l'ENSTA Bretagne, qui m'a dirigé vers des lectures complémentaires commes les cours de l'université de Berkeley où ceux de David Silver, professeur d'informatique à l'université de Londres et un cours en ligne sur Hugging Face, une plateforme web dédiée à l'apprentissage de l'intelligence artificielle.

III.1.2 Formation Pratique via Hugging Face

Le cours en ligne sur Hugging Face, composé de douze leçons mêlant théorie et pratique, m'a permis de mettre en œuvre progressivement les concepts théoriques que j'avais appris. L'un des points forts de cette formation était l'utilisation de la bibliothèque *Gym*, une plateforme qui simule des environnements pour l'apprentissage par renforcement. J'ai ainsi pu entraîner des agents dans des environnements variés, allant de jeux vidéo tels que *Space Invaders* et *Doom* à des simulations robotiques plus complexes, comme le contrôle d'un bras robotique et le maintien d'un pendule inversé en équilibre.

Ces exercices m'ont permis d'appréhender des notions clés comme la fonction de récompense, qui guide l'agent à optimiser ses actions en fonction des résultats obtenus. L'ajustement précis de cette fonction est crucial pour façonner le comportement de l'agent tout au long de son apprentissage.

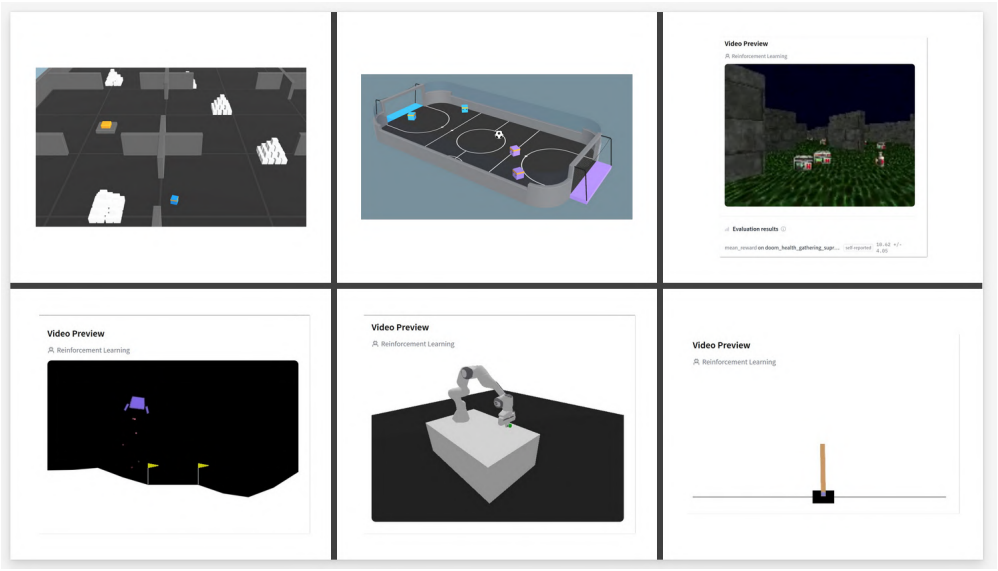

FIGURE III.1 – Exemple de différentes simulations de résolution de jeu par RL que j'ai effectuées durant les cours sur la plateforme Hugging Face

Parmi les simulations réalisées (figure ci-dessus), on retrouve des scénarios où il fallait :
— Entraîner un agent à faire tomber une pyramide du cube orange.
— Entraîner des agents à jouer au football contre des adversaires, eux-mêmes pilotés par IA.
— Survivre le plus longtemps possible en collectant des kits de soin dans une version de *Doom*.

— Entraîner un bras robotique à atteindre des positions précises.
— Maintenir un bâton en équilibre en déplaçant un véhicule horizontalement.

Enfin, le cours se terminait par une étude approfondie de l'algorithme *Proximal Policy Optimization* (PPO), un algorithme de deep reinforcement learning réputé pour sa stabilité et son efficacité. J'ai pu comprendre comment cet algorithme limite les ajustements brusques des actions de l'agent, garantissant ainsi une convergence stable vers des solutions optimales.

III.2 Bilan de l'Apprentissage et Transition vers la Pratique

Mon apprentissage de l'algorithme *Proximal Policy Optimization* (PPO) a marqué un tournant dans ma formation en intelligence artificielle. Cet algorithme, reconnu pour sa stabilité et son efficacité, constitue un outil clé dans les systèmes d'apprentissage par renforcements avancés. Son utilisation, via la bibliothèque *Stable Baselines3*, m'a permis d'acquérir une maîtrise fine des mécanismes d'entraînement des agents, notamment en ajustant les hyperparamètres de manière à optimiser leur performance.

Cette période d'apprentissage, à la fois dense en théorie et riche en exercices pratiques, m'a préparé à la prochaine étape de mon stage : l'application directe de ces concepts sur des modèles robotiques. Loin de se limiter à une simple compréhension des algorithmes, cette formation m'a permis de franchir le cap vers une mise en œuvre concrète dans des environnements simulés puis réels. Grâce à cette base solide, j'étais enfin prêt à intégrer l'apprentissage par renforcement dans des contextes robotiques complexes, tels que le contrôle de bras robotisés ou de systèmes à plusieurs degrés de liberté.

Dans la section suivante, je détaillerai comment ces compétences théoriques ont été mises en pratique lors de mon travail au laboratoire, en particulier dans le cadre du contrôle et de l'optimisation des modèles robotiques tels que le bras Franka Emika Panda et le xArm6.

FIGURE III.2 – Certificat obtenu à la fin de ma formation en ligne

III.3 Mise en pratique sur des modèles robotiques existant et premières expérimentations

III.3.1 Tentative d'application de mon apprentissage sur un modèle complexe, cas du Robot humanoïde

Après avoir suivi une formation en ligne et exploré diverses ressources scientifiques, je me suis senti prêt à appliquer les connaissances acquises à un projet concret et utile pour le laboratoire. Mon tuteur m'a alors proposé de consulter des références scientifiques spécifiques au contrôle de robots humanoïdes. Le premier article [ALSR19] traitait de l'apprentissage d'un robot nano pour jouer au football, tandis que le second [SBM+22] était axé sur l'entrainement à la marche bipédique d'un robot humanoïde. Ce dernier article, notamment rédigé par le Dr Singh, expert en contrôle par renforcement des robots humanoïdes et membre de notre équipe de recherche, m'a permis d'approfondir mes connaissances sur des applications pratiques.

Grâce à ma formation théorique, j'ai pu comprendre ces différentes ressources et les diverses stratégies de recherche, même si les articles étaient plus complexes que ce que j'avais étudié. J'ai eu l'opportunité d'effectuer des simulations basées sur le code du dernier article, me permettant ainsi d'entraîner mon premier robot humanoïde à suivre un champ de marche prédéfini, suivant la structure présenté dans la figure III.3.

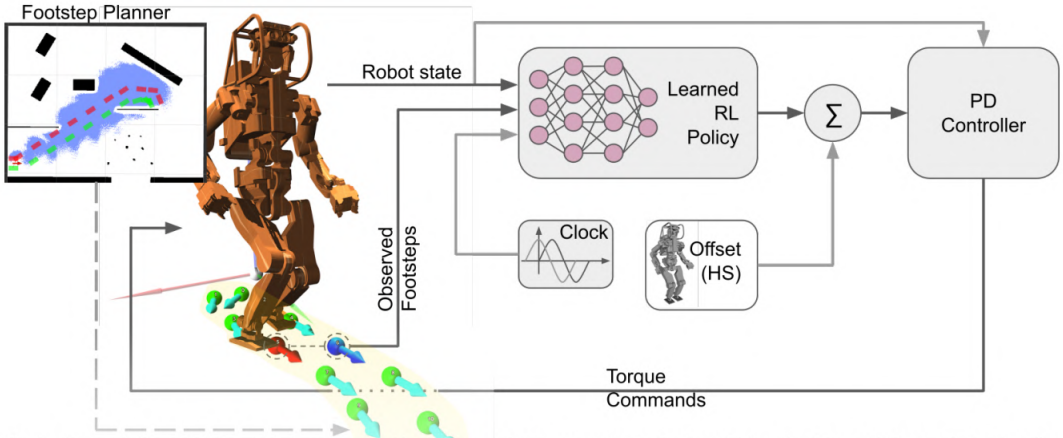

Fig. 2. **Proposed hierarchical control structure.** The high-level controller is represented by the learned RL policy. The predictions made by RL policy π are added to the neutral motor-positions, corresponding to the "half-sitting" (HS) posture of HRP-5P, and then sent to the PD control loop. The input to the policy are the next two planned footsteps ($\mathbf{T}_1$ and $\mathbf{T}_2$), the clock signal, and the robot state. Footstep planning is performed by conventional methods.

FIGURE III.3 – Schéma du fonctionnement du controle par RL de la marche bipédique d'un robot humanoïde [SBM+22]

Dans cette structure, les prédictions de la politique RL sont intégrées aux positions motrices neutres du robot, qui correspondent à une posture spécifique appelée « demi-assise » (HS). Une fois que ces ajustements sont effectués, les informations sont envoyées à une boucle de contrôle proportionnelle-dérivée (PD). Cette boucle est responsable de l'exécution précise des mouvements en réponse aux commandes du niveau supérieur. Les entrées fournies à la politique RL comprennent les deux prochaines étapes prévues pour le robot (T1 et T2), un signal d'horloge pour le timing des mouvements, ainsi que l'état actuel du robot (comme sa position et sa vitesse). Pour déterminer où le robot doit aller,

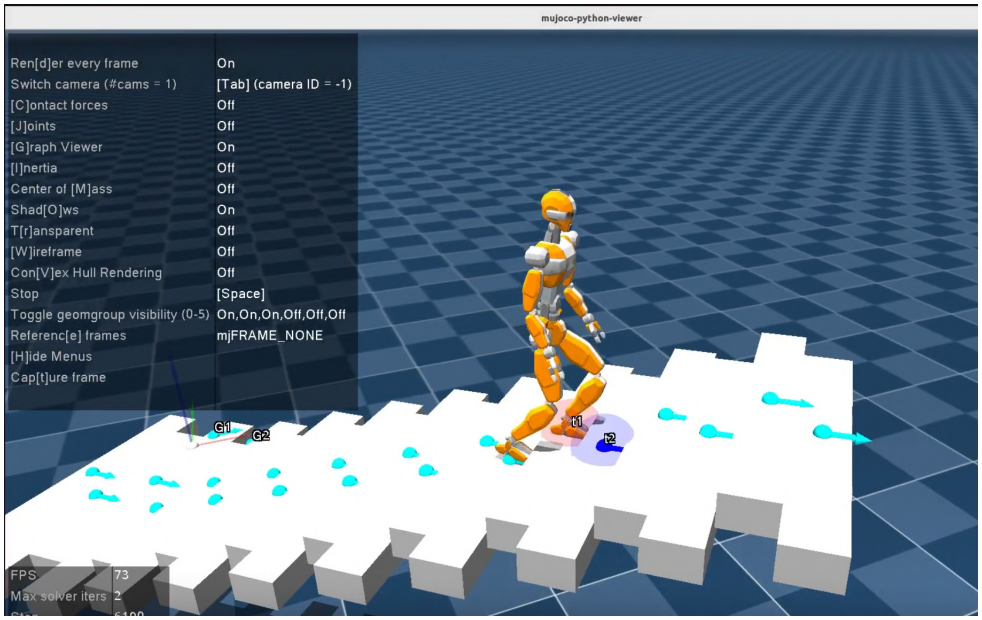

FIGURE III.4 – Simulation MuJoCo d'une marche bipédique d'un robot humanoïde entrainé par RL

la planification des pas s'effectue à l'aide de méthodes conventionnelles, ce qui permet de garantir que le robot se déplace de manière efficace et stable dans son environnement.

Cependant, cette approche s'est avérée insuffisante, car, en apprentissage par renforcement, l'objectif est de minimiser au maximum l'intervention humaine dans la planification du contrôle. L'un des principaux objectifs de notre groupe de recherche était de supprimer cette planification, permettant ainsi au robot de déterminer lui-même le chemin optimal pour accomplir la tâche.

Après plusieurs semaines de réflexions et d'essais de modifications sur le code, j'ai réalisé que ce sujet était plus complexe que prévu et que la compréhension du code présentait également des défis significatifs. Cependant, ces petites expérimentations m'ont permis de me familiariser avec le logiciel de simulation MuJoCo. Ce dernier est particulièrement reconnu dans le domaine de la robotique pour sa capacité à simuler des environnements physiques complexes et à modéliser des mouvements réalistes, ce qui est fondamental pour l'apprentissage par renforcement. Mon expérience avec MuJoCo a été essentielle, car c'est le logiciel que j'ai utilisé tout au long de mon stage pour réaliser toutes mes simulations.

Ainsi, après un mois de stage, j'ai opéré une redirection vers un sujet qui pourrait réellement apporter une valeur ajoutée au laboratoire en termes de recherche. Cela m'a permis d'éviter de passer le reste de mon stage à déchiffrer le code de simulation du robot humanoïde sans réelle valeur ajoutée, tant pour moi que pour le laboratoire.

III.3.2 Découverte d'un modèle plus simple, le bras robotique Franka Emika Panda

Durant mon apprentissage sur la plateforme Hugging Face, j'ai dû entraîner un bras robotique afin qu'il remplisse différentes tâches, comme atteindre une position précise. Étant donné qu'il y avait un bras robotique au laboratoire, mon tuteur et moi avons convenu qu'il serait pertinent de travailler dessus et d'appliquer concrètement les méthodes

apprises lors de ce cours. Je me suis donc concentré sur le contrôle de ce bras.

Approche et inspiration

Pour ce faire, je me suis inspiré des travaux de Zichun Xu et al. intitulés *Open-Source Reinforcement Learning Environments Implemented in MuJoCo with Franka Manipulator* [XLY+23], qui traitent du contrôle par apprentissage par renforcement du bras robotique Franka Emika Panda. Ce bras, composé de 7 axes, est très utile dans le domaine industriel. Cet article présentait leur recherche pour contrôler ce bras par apprentissage par renforcement et proposait également un lien vers un dépôt GitHub contenant le code des fonctions du robot simulé sur le logiciel MuJoCo. Cet article a été essentiel pour commencer ma nouvelle tâche.

Cependant, après avoir lu cet article, j'ai remarqué que les chercheurs entraînaient la politique de leur agent pour qu'elle renvoie une position à atteindre pour l'effecteur final. Cette méthode reposait sur des calculs mathématiques basés sur le modèle du robot, pour déterminer comment actionner chaque actionneur afin d'atteindre cette position. Cette approche ne nous satisfaisait guère, car l'intérêt principal de l'apprentissage par renforcement est justement d'éviter tout calcul mathématique basé sur le modèle du robot. J'ai donc repris l'ensemble du code fonctionnel du robot pour contrôler directement les positions des actionneurs. J'ai également implémenté des algorithmes d'entraînement afin d'entraîner le bras robotique et d'observer son comportement en simulation.

Tests et résultats

Pour tester le bon fonctionnement ainsi que l'efficacité de mes fonctions, j'ai entraîné mon bras robotique sur une tâche simple : "Reach", soit atteindre un point généré aléatoirement dans l'espace de simulation. Dans ce modèle, la fonction de récompense était l'opposé de distance entre l'objectif et la position de l'effecteur final. L'objectif de ce travail était de retrouver les mêmes résultats que ceux énoncés dans le papier de recherche, mais en contrôlant uniquement la position des actionneurs et en minimisant les calculs du modèle dynamique. Minimiser les calculs était une priorité, même si j'ai dû recourir à ces calculs pour déterminer la position de l'effecteur final afin de calculer la récompense. Cependant, ce recours peut être évité si on utilise un capteur qui nous renvoie directement la distance entre l'effecteur final et l'objectif.

Les figures ci-dessous illustrent l'efficacité de l'entraînement par apprentissage par renforcement. La première figure montre les différentes positions prises par le robot en fonction des pas de simulation, tout en contrôlant les actionneurs avec des positions aléatoires. Ces simulations me permettaient de savoir si le robot fonctionnait correctement avec les bonnes entrées et sorties. Cependant, elles témoignent également de l'impact de l'entraînement sur le contrôle. La deuxième figure montre quant à elle la distance entre l'objectif et l'effecteur final une fois que la politique de contrôle du bras est entraînée. Pour ce test très simple, j'ai entraîné mon agent, le bras robotique, sur un million de pas.

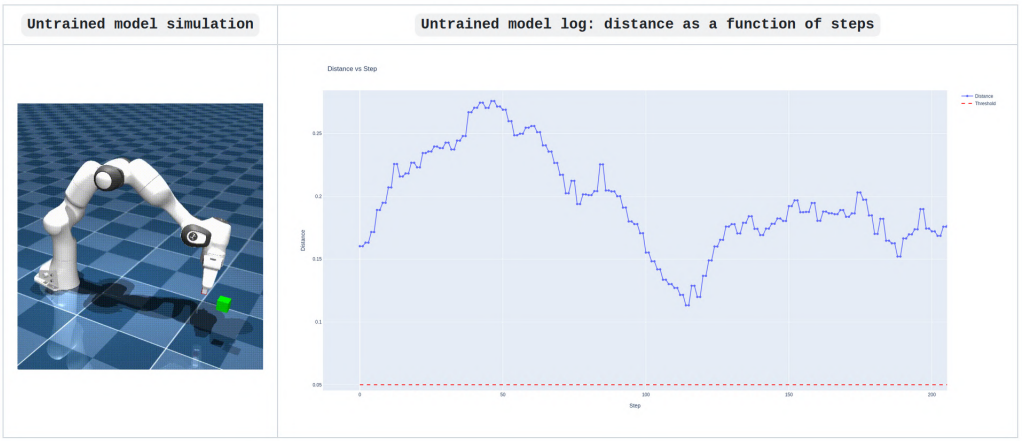

FIGURE III.5 – Simulation MuJoCo du bras Panda non-entrainé, Graphique de la distance à l'objectif en fonction des pas de simulation

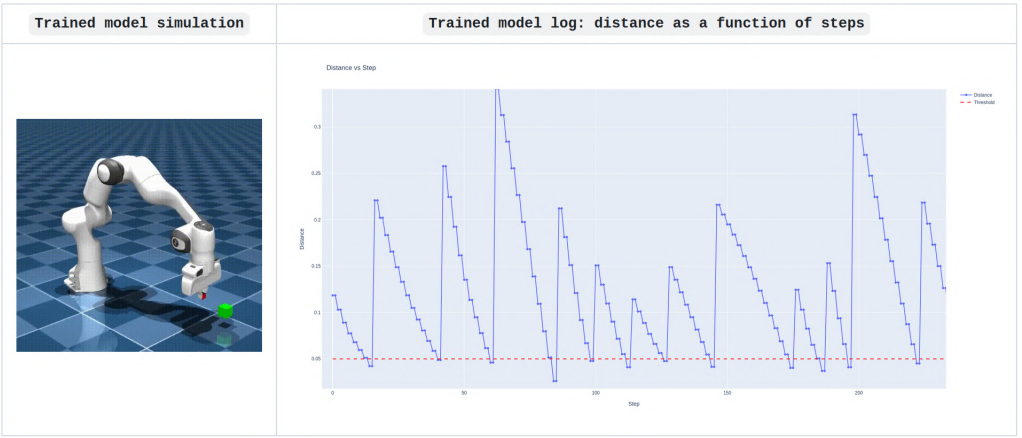

FIGURE III.6 – Simulation MuJoCo du bras Panda entrainé, Graphique de la distance à l'objectif en fonction des pas de simulation

Ainsi, lors de mes présentations, ce modèle très simple a servi d'exemple afin d'expliquer à tous les membres du laboratoire et aux personnes s'intéressant à mon sujet tous les concepts de l'apprentissage par renforcement. Par la suite, j'ai également effectué quelques tests pour améliorer mon modèle.

Améliorations et objectifs futurs

J'ai essayé différentes stratégies pour que le robot reste à la position voulue, sans se réinitialiser. Durant l'entraînement par renforcement, cette réinitialisation est essentielle, car elle permet de terminer un épisode et de compter l'ensemble des récompenses, afin de déterminer si les positions choisies sont intéressantes pour atteindre l'objectif. J'ai donc exploré comment supprimer cette réinitialisation sans perturber le modèle d'entraînement. J'ai identifié deux solutions : la première consiste à maintenir la position des actionneurs une fois l'objectif atteint, tandis que la deuxième envisage de ne pas considérer l'atteinte de l'objectif comme une condition terminale.

Cependant, toutes ces pratiques sur le bras Panda n'étaient qu'un moyen pour moi de

mettre en application mes enseignements. Le réel objectif était surtout de contrôler le bras du laboratoire, qui est un bras xArm6 avec 6 actionneurs, contre 7 pour le Panda. La suite de mon travail a donc consisté à reproduire ce que j'avais fait avec le bras Panda, mais cette fois-ci en partant de zéro, sans modèle 3D.

III.4 Définition du modèle final et simulation

III.4.1 Elaboration du modèle 3D du robot xArm6

Importance du modèle 3D dans l'apprentissage par renforcement

Avant d'effectuer des manipulations sur le robot réel, il est nécessaire de mener des tests en simulation. En effet, dans le domaine du contrôle de robots, les simulations jouent un rôle fondamental, car elles permettent d'effectuer de nombreux essais à un coût réduit, tant sur le plan financier que matériel ou humain. À ce stade de mon stage, le nouveau post-doctorant du laboratoire devait également travailler sur le bras robotique.

Au-delà du simple contrôle des robots, la simulation informatique est essentielle dans le cadre de l'apprentissage par renforcement. Cette technique repose sur le principe de répétition pour s'améliorer dans l'exécution d'une tâche. Ainsi, dans l'apprentissage par renforcement, il est primordial d'avoir un modèle 3D qui se rapproche le plus possible de la réalité. Cela permet de garantir que les modifications apportées à la politique, suite aux différentes simulations, soient également applicables au robot réel.

La stratégie la plus efficace pour contrôler un robot par apprentissage par renforcement consiste à utiliser une politique entraînée sur un jumeau numérique. Toutefois, il est important de noter qu'une simulation présente généralement moins de perturbations qu'un environnement réel. Idéalement, il serait préférable d'entraîner la politique directement sur le robot réel. Cependant, dans un environnement réel, la durée des épisodes d'entraînement est bien plus longue que dans une simulation, ce qui rend l'entraînement complet beaucoup plus lent. Pour obtenir des résultats optimaux, il est nécessaire de réaliser idéalement des millions de pas de simulation.

De plus, les algorithmes de Proximal Policy Optimization (PPO) offrent la possibilité de multiplier le nombre d'agents. Ainsi, plutôt que d'entraîner une politique sur un seul bras robotique, il est possible de l'entraîner simultanément sur plusieurs bras, renforçant ainsi l'efficacité de la politique.

Pour toutes ces raisons, et dans l'objectif d'avancer durant mon stage, il m'était essentiel de concevoir un jumeau numérique du bras xArm6 présent au laboratoire.

Conception du modèle 3D du robot xArm6

En consultant l'article [XLY+23], qui m'avait précédemment aidé à comprendre le contrôle du bras Franka Emika Panda, j'ai découvert que le groupe avait utilisé MuJoCo Ménagerie [ZTM22], un dépôt GitHub proposant des modèles 3D directement exploitables dans l'environnement MuJoCo. Ce repertoire contient non seulement les fichiers 3D des pièces de robots, mais aussi les documents XML nécessaires à leur assemblage, à la définition des liaisons entre ces pièces, ainsi qu'à la création de scènes pour l'utilisation des robots.

Cependant, il est à noter que, bien que le repertoire github offre des modèles de divers robots, le modèle du bras xArm6 n'y figurait pas. À la place, j'ai trouvé le modèle du robot xArm7, également conçu par UFactory, qui se distingue par ses sept actionneurs au lieu

des six présents sur le xArm6. Comme le montre la figure III.7, la composition des deux modèles étant différente, je ne pouvais pas utiliser les pièces du bras xArm7 pour créer le modèle du bras xArm6.

FIGURE III.7 – Comparaison des bras robotiques xArm6 (gauche) et xArm7 (droite)

Pour la modélisation 3D, je me suis appuyé sur des fichiers disponibles en ligne. Ces fichiers, créés à l'origine sur SolidWorks, ont été convertis pour une utilisation dans Autodesk Inventor. Ensuite, j'ai adapté chaque pièce pour garantir la compatibilité avec MuJoCo. Cette adaptation a impliqué des ajustements concernant les tailles, les référentiels, ainsi que les positions et orientations de chaque pièce.

Une fois les fichiers modifiés et convertis en .stl, j'ai abordé la création du fichier .xml nécessaire à l'assemblage des pièces, à la gestion des dépendances et des liaisons. Dans ce processus, l'accès au fichier .xml du bras xArm7, disponible dans MuJoCo Ménagerie, m'a été d'une grande aide. Cela m'a permis de mieux comprendre le fonctionnement de ces fichiers et d'adapter le mien pour concevoir un modèle 3D fonctionnel du xArm6.

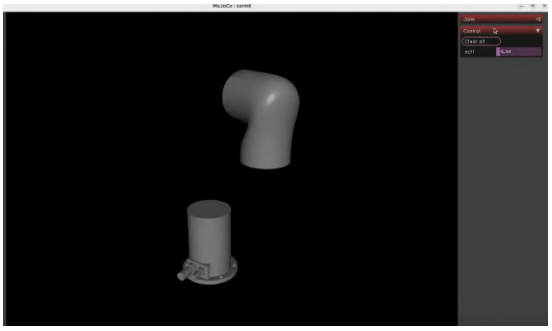

FIGURE III.8 – Modèle 3D 2 pièces et 1 liaison du bras xArm6 sur MuJoCo

FIGURE III.9 – Modèle 3D final du bras xArm6 sur MuJoCo

FIGURE III.10 – Évolution de la conception du modèle 3D du bras xArm6

III.4.2 Implémentation des algorithmes et fonctions de contrôle du robot

Une fois le modèle 3D du bras robotique créé et entièrement contrôlable via le MuJoCo Viewer, où il était possible de modifier manuellement la position des actionneurs, il était nécessaire de développer des scripts pour contrôler l'environnement du robot à partir d'un code Python. Les manipulations que j'avais effectuées sur les codes du robot Franka Emika Panda m'ont été d'une grande aide dans cette phase, notamment pour comprendre la dépendance entre différentes classes et organiser mes codes en conséquence.

En effet, mon code, qui permet de contrôler le fonctionnement basique du robot sans implémentation de scène ou d'algorithme de régulation, devait hériter de la classe d'environnement robotics de MuJoCo, *MujocoRobotEnv*. Cela me permettait d'exploiter le modèle dans la simulation, par exemple en modifiant la position des actionneurs, en récupérant leur état ou encore en obtenant la position de l'effecteur final dans l'espace de simulation. Bien que cette étape n'ait pas été la plus complexe, elle a pris du temps, car il m'a fallu explorer en détail la documentation des classes robotiques de MuJoCo pour adapter correctement mon nouveau modèle 3D aux fonctionnalités de MuJoCo.

Comme pour le précédent bras robotique, cette étape s'est conclue par une simulation où les positions des actionneurs étaient générées aléatoirement, sans implémentation d'algorithme de contrôle. Cela m'a permis de vérifier que le contrôle de base fonctionnait correctement en simulation.

La figure suivante montre le résultat de cette simulation pour le bras xArm6 non entraîné, où la distance à l'objectif est représentée en fonction des pas de simulation.

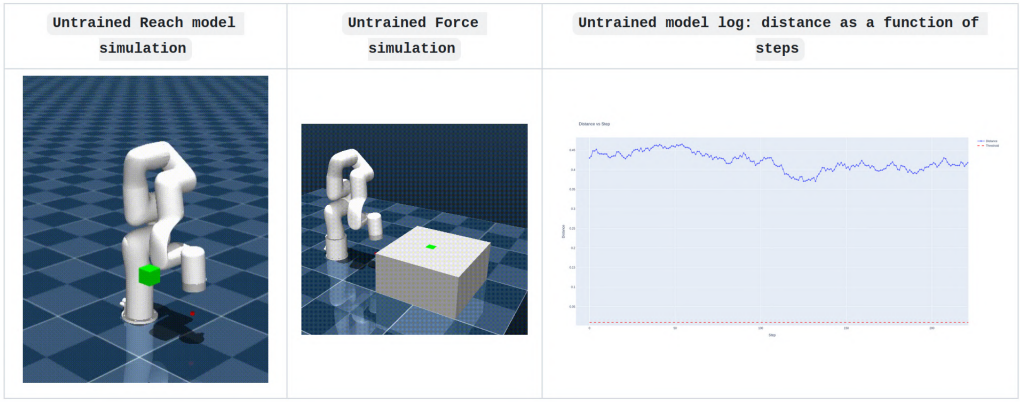

FIGURE III.11 – Simulation MuJoCo du bras xArm6 non entraîné : Graphique de la distance à l'objectif en fonction des pas de simulation

III.4.3 Création et test du robot sur deux tâches : 'Reach' et 'Reach and Force'

La figure III.11 illustre le comportement du robot non entraîné dans deux environnements distincts, chacun représentant une tâche que nous souhaitions que le robot accomplisse. Le concept de "tâche" est un élément clé en apprentissage par renforcement, car il définit la mission que l'on souhaite que le robot réalise. En termes de programmation, cela se traduit par la création d'un nouveau fichier qui hérite de la classe définie précédemment, et qui introduit les fonctions essentielles au *reinforcement learning*, comme la fonction de récompense et la fonction d'observation. Cette dernière permet de fournir à la politique des informations précises sur l'état du robot et de son environnement à chaque instant.

Dans le cadre de cette étude, j'ai implémenté deux tâches pour le robot xArm6 :

— La tâche *Reach*, identique à celle utilisée précédemment pour le bras robotique Panda,

— Et la tâche *Reach and Force*, qui combine deux objectifs : atteindre une position donnée, puis y exercer une force spécifique.

La tâche *Reach*

Comme mentionné plus tôt, cette tâche est exactement la même que celle utilisée pour le bras Panda. Cependant, avec l'introduction des notions de fonction de récompense et d'observation, nous pouvons désormais l'expliquer plus en détail.

Pour chaque tâche, il est nécessaire de définir ces notions, car la réussite d'un bon entraînement dépend largement du choix de ces paramètres. Pour cette tâche, j'ai choisi la structure suivante :

$$\text{Observation} = \begin{cases} \text{"observation"} & : (\text{ee\_position, ee\_velocity}) \\ \text{"achieved\_goal"} & : (\text{ee\_position}) \\ \text{"desired\_goal"} & : (\text{goal\_position}) \end{cases} \tag{III.1}$$

$$\text{reward} = -\,\text{distance\_target\_ee} \tag{III.2}$$

Dans cette configuration, l' "observation" regroupe les informations relatives au robot, telles que la position et la vitesse de l'effecteur final. L' "achieved goal" correspond à la position atteinte par l'effecteur final à la fin de chaque étape, tandis que le "desired goal" représente la position de l'objectif visé. Ces deux dernières valeurs permettent de calculer la distance entre l'effecteur final et l'objectif, ce qui permet de déterminer si l'objectif est atteint. Comme précédemment, la fonction de récompense est définie comme l'opposé de cette distance, afin que la récompense devienne plus élevée au fur et à mesure que le robot se rapproche de l'objectif. L'objectif est donc de rendre cette valeur aussi faible que possible, voire nulle.

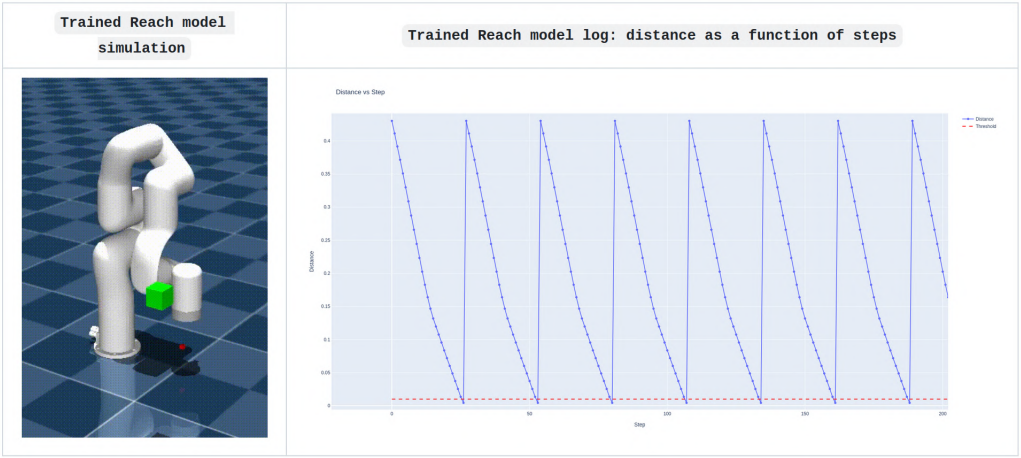

FIGURE III.12 – Simulation MuJoCo du bras xArm6 entraîné pour la tâche *Reach* : Graphique de la distance à l'objectif en fonction des pas de simulation

Après l'entraînement, les résultats obtenus sont identiques à ceux du bras Panda, démontrant ainsi l'efficacité de l'apprentissage.

Cette implémentation m'a également permis de réaliser la puissance de l'apprentissage par renforcement. En effet, j'ai pu réutiliser exactement le même code que pour le bras Panda, et en seulement un entraînement, j'ai obtenu des résultats similaires. Bien que les politiques d'entraînement ne soient pas interchangeables entre les deux robots, le processus d'entraînement, lui, ne pose aucun problème. Cet exemple met en lumière l'avantage

de l'apprentissage par renforcement par rapport aux méthodes basées sur des modèles physiques. Une fois les algorithmes de contrôle correctement définis, il devient possible d'entraîner d'autres robots, même avec des configurations d'actionneurs différentes, en seulement quelques heures. En revanche, avec les approches basées sur des modèles, si les calculs de cinématique inverse ne sont pas déjà disponibles dans une bibliothèque ou directement implémentés, le contrôle devient beaucoup plus complexe, voire impossible.

La tâche *Reach and Force*

Bien que la tâche *Reach* permette de se familiariser avec les concepts de l'apprentissage par renforcement, elle reste relativement simple en simulation et n'apporte pas de réelles perspectives pour l'équipe de recherche sur l'amélioration du contrôle des robots via cette méthode. Leur principal intérêt réside dans le contrôle en environnement multi-contact, un aspect important pour qu'un robot humanoïde puisse interagir efficacement avec son environnement. En particulier, l'un des objectifs de l'équipe était de permettre à un robot de se déplacer en s'appuyant sur des obstacles tels que des murs.

Ainsi, sous la supervision du Dr Marwan Hamze, j'ai travaillé sur une tâche combinant la mise en position de l'effecteur final et la gestion de contacts avec une surface, comme un mur ou une table. Le véritable défi résidait dans la conception d'une fonction de récompense capable de gérer ces deux sous-tâches simultanément. Ce travail était d'autant plus complexe qu'il n'existe, à ce jour, aucune publication scientifique officielle fournissant une méthodologie claire sur l'association de plusieurs tâches en renforcement, notamment en ce qui concerne la gestion des consignes de force.

Cette recherche a des implications intéressantes, car actuellement, le contrôle par renforcement est généralement limité à une tâche unique. Si l'on parvient à combiner plusieurs tâches, cela pourrait considérablement étendre les possibilités, par exemple en permettant à un robot de réaliser diverses actions tout en se déplaçant. L'impact potentiel de cette avancée serait de permettre un contrôle plus flexible et adaptable, avec des applications variées comme l'association avec des machines à états ou le contrôle multi-objectifs.

Pour mes tests, j'ai utilisé la structure suivante pour les observations :

$$\text{Observation} = \begin{cases} \text{"observation"} & : (\text{ee\_position, ee\_velocity, ee\_force}) \\ \text{"achieved\_goal"} & : (\text{ee\_position, ee\_force}) \\ \text{"desired\_goal"} & : (\text{goal\_position, goal\_force}) \end{cases} \tag{III.3}$$

Algorithme de Récompense

Input: ee_position, ee_velocity, ee_force, goal_position, goal_force
Output: reward
distance ← calculate_distance(ee_position, goal_position);
contact ← check_contact(ee_force);
if *distance < threshold* ***and*** *contact > 0* **then**
  reward ← $-\beta \times$ force_error;
else
  reward ← $-\alpha \times$ *distance*;
end

Pour cette tâche, j'ai ajouté des informations supplémentaires concernant les forces exercées, car le succès est défini à la fois par l'atteinte de la position de l'objectif et l'application d'une force précise. Le véritable défi a été de trouver une fonction de récompense adéquate. Dans cet algorithme, j'utilise deux indicateurs : la distance à l'objectif et le contact avec l'obstacle (le mur ou la table). Si l'effecteur final n'a pas atteint la position cible et n'est pas en contact, la récompense est simplement basée sur la distance, comme dans la tâche *Reach*. Sinon, la récompense est calculée en fonction de l'erreur de force entre la consigne et la force appliquée.

Un point clé dans cet algorithme réside dans le choix des coefficients proportionnels α et β, qui ajustent le calcul de la récompense. Lors de mes expériences, j'ai rencontré un problème où le bras, après apprentissage, tendait à rester à la limite de la condition "sinon", car cela lui offrait en moyenne une meilleure récompense. Pour contourner cela, j'ai ajusté β pour inciter le robot à réaliser les deux sous-tâches plutôt que d'éviter le contact.

Grâce à cette structure, j'ai obtenu la simulation suivante :

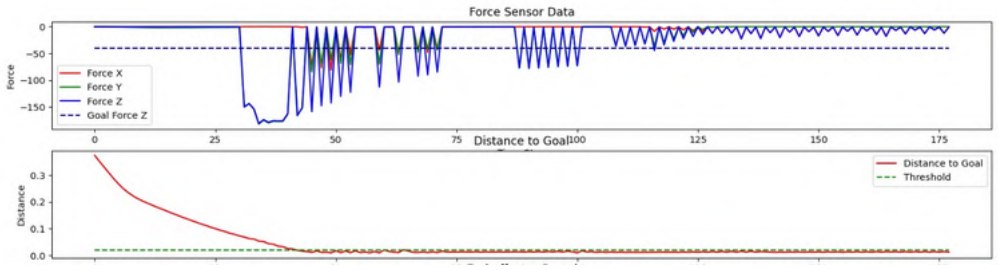

FIGURE III.13 – Simulation MuJoCo du bras xArm6 entraîné pour la tâche *Reach and Force* : Graphique de la force et de la distance à l'objectif en fonction des pas de simulation

Sur cette figure, on observe que le bras robotique atteint facilement la position de l'objectif. Cependant, il échoue à appliquer une force constante de -50 N sur l'axe z. On remarque quelques pics indiquant que le robot tente d'appliquer cette force, mais ces tentatives restent peu concluantes.

Cette expérience m'a permis de constater les limites et les difficultés du contrôle par apprentissage par renforcement, notamment son aspect probabiliste et aléatoire. Contrairement à un contrôle déterministe, il est difficile d'exercer un contrôle total sur le comportement du robot, ce qui complique considérablement le débogage. Dans cet exemple, de nombreux facteurs pourraient expliquer les résultats obtenus : la définition des fonctions, le calcul des forces dans MuJoCo, ou encore les hyperparamètres de l'algorithme d'apprentissage, qui pourraient ne pas être optimaux.

Le temps limité de mon stage ne m'a pas permis d'aller au-delà de ces essais pour cette tâche.

III.5 Expérimentation sur robot réel

III.5.1 Mise en place et résolution des problèmes pour une implémentation réelle

Au-delà du travail en simulation, l'objectif principal de mon stage était d'implémenter le contrôle par apprentissage par renforcement sur un robot réel, en l'occurrence le bras

robotique xArm6.

La principale difficulté technique résidait dans la réécriture complète des environnements, en conservant la structure des codes utilisés pour les simulations, mais sans l'utilisation des bibliothèques de MuJoCo. À la place, j'ai dû recourir à l'API du bras réel. Par ailleurs, il était essentiel que ces nouveaux environnements soient compatibles avec la bibliothèque Gym, car c'est à travers elle que mes environnements pouvaient être exploités par les algorithmes d'intelligence artificielle.

Cette transition vers le contrôle d'un robot réel m'a obligé à repenser l'architecture de mon répertoire de code afin de permettre une distinction claire entre les environnements simulés et réels. Cela devait être fait de manière à ce que tout utilisateur, y compris moi-même, puisse s'y retrouver facilement. De plus, au cours de cette étape, j'ai découvert qu'une des articulations du modèle tournait dans le sens inverse de celle du robot réel. J'ai donc dû adapter entièrement mon modèle 3D afin de l'aligner correctement avec les mouvements du bras physique, et ainsi éviter des problèmes de dépassement de plage (*overrange*) et potentiellement dangereux lors des expérimentations.

III.5.2 Test de contrôle du robot sans régulation via les scripts Python

Le bras robotique fourni par l'entreprise UFactory est équipé d'un logiciel permettant un contrôle manuel. Cependant, mon objectif était de le contrôler directement à l'aide de scripts Python. Cela m'a amené à explorer l'API du robot pour identifier les fonctions capables de remplacer celles de la bibliothèque MuJoCo utilisées pour les simulations. Ce travail a nécessité une exploration approfondie de la documentation, à la recherche des fonctions pertinentes, et leur implémentation dans mon propre code.

À l'issue de ce processus, j'ai pu réaliser un test préliminaire pour vérifier si le contrôle de base, sans régulation, fonctionnait. Ce test consistait à générer des mouvements aléatoires des actionneurs en boucle, comme lors des simulations.

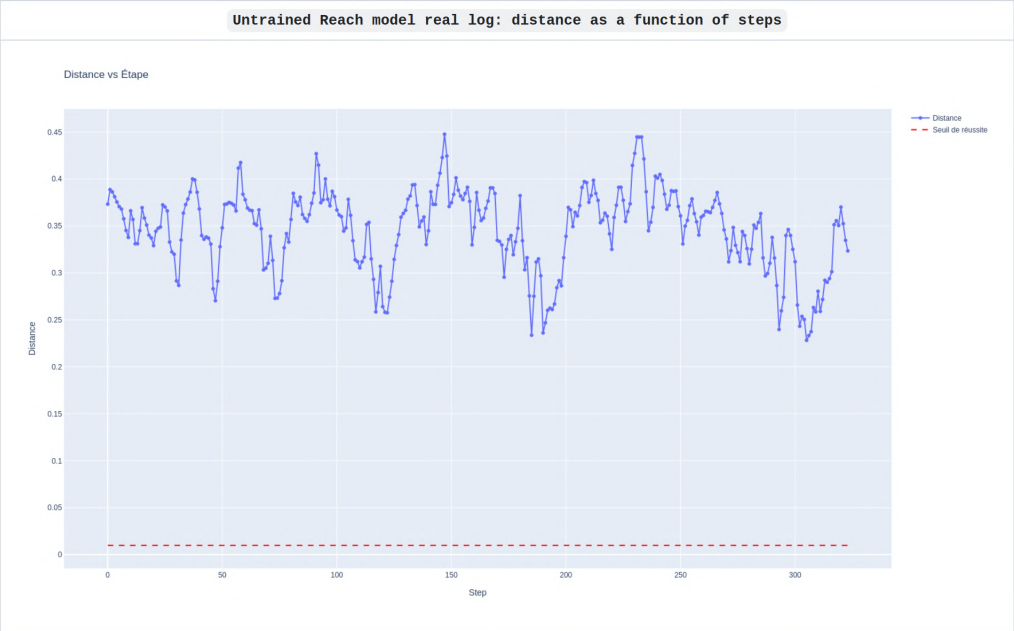

FIGURE III.14 – Expérimentation réelle du bras xArm6 non-entraîné : Graphique de la distance à l'objectif en fonction des pas d'expérience

III.5.3 Test et implémentation de la tâche *Reach* sur le robot réel

Comme le montre la figure III.14, le robot prend bien des positions aléatoires sans rencontrer de problèmes majeurs. Ces résultats confirment que mes scripts de contrôle fonctionnent correctement. Je suis donc passé à l'implémentation des tâches de contrôle.

Pour commencer, j'ai choisi d'implémenter la tâche *Reach*, car elle est la plus simple et ne m'a jamais posé de difficulté en simulation. J'ai simplifié cette tâche en fixant un objectif statique, identique à celui utilisé dans les tests simulés. Une fois l'apprentissage terminé et la politique d'entraînement jugée satisfaisante, je l'ai sauvegardée pour l'utiliser sur le robot réel.

Après avoir lancé l'expérience sur le bras xArm6, voici les résultats obtenus :

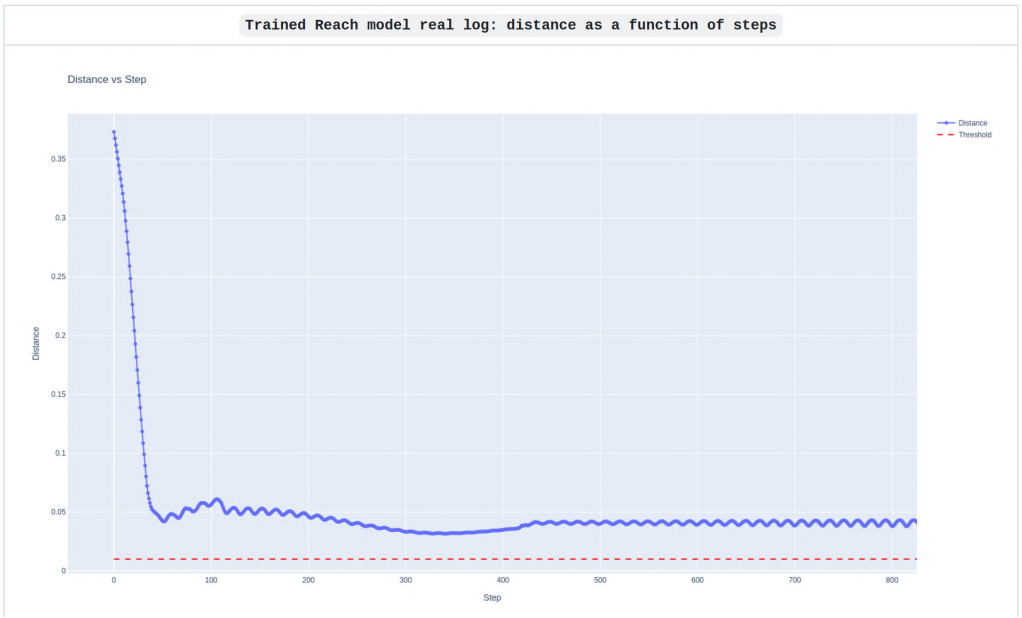

FIGURE III.15 – Expérimentation réelle du bras xArm6 entraîné pour la tâche *Reach* : Graphique de la distance à l'objectif en fonction des pas d'expérience

À partir de ces résultats, on observe une erreur d'environ 4 cm entre la position de l'effecteur final et le seuil de l'objectif.

Identifier précisément l'origine de cette erreur et trouver une solution pour la réduire reste un défi. Une option pourrait être d'augmenter le nombre de pas de temps (*timesteps*) durant l'apprentissage, afin de permettre au modèle d'affiner ses actions.

Cependant, en raison de contraintes de temps lors de mon stage, je n'ai pas pu approfondir cette problématique ni la résoudre entièrement.

Chapitre IV

Conclusion

Mon stage au sein du laboratoire Yoshida a été une expérience formatrice à la fois sur le plan scientifique et humain. En travaillant sur le contrôle par apprentissage par renforcement d'un bras robotique, j'ai pu acquérir des compétences solides dans l'intégration de l'intelligence artificielle à des environnements robotiques réels. Le passage des simulations MuJoCo à la manipulation directe du bras xArm6 m'a offert des perspectives concrètes sur les défis et les subtilités liés à l'expérimentation sur des plateformes physiques. De plus, l'adaptation de mes algorithmes à ces systèmes a représenté une occasion unique de progresser dans la compréhension de l'apprentissage par renforcement appliqué à la robotique.

Durant ce stage, j'ai également créé un répertoire GitHub [Rom24] regroupant l'ensemble de mes recherches et des codes développés. Ce dépôt documente les différentes étapes du projet, de la simulation à l'expérimentation sur le robot réel, et pourra servir de référence pour les futurs chercheurs et stagiaires du laboratoire.

Au-delà des aspects techniques, cette immersion dans un environnement multiculturel, avec ses propres défis linguistiques et sociaux, m'a permis de mieux appréhender la coopération internationale en recherche. Mon intégration au sein de l'équipe japonaise, ainsi que les échanges réguliers avec des chercheurs de divers horizons, ont été des éléments clés de mon développement personnel. Les difficultés rencontrées, notamment en matière de communication, ont été surmontées grâce à l'entraide et au soutien constant de mes encadrants et collègues.

Ce stage a ainsi confirmé mon intérêt pour la recherche en robotique et en intelligence artificielle, tout en renforçant ma capacité à travailler de manière autonome et à résoudre des problèmes complexes dans des contextes interculturels. Il constitue un socle solide pour la poursuite de mes études et de mes recherches dans ces domaines passionnants.

Bibliographie

[ALSR19] Miguel Abreu, Nuno Lau, Armando Sousa, and Luis Paulo Reis. Learning low level skills from scratch for humanoid robot soccer using deep reinforcement learning. In *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, Porto, Portugal, 2019. IEEE. 14

[Kea01] Shuuji Kajita and et al. The 3d linear inverted pendulum mode: A simple modeling for a biped walking pattern generation. In *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the Next Millennium (Cat. No. 01CH37180)*, volume 1. IEEE, 2001. 6

[Kea05] Shuuji Kajita and et al. *Introduction to Humanoid Robotics*. Springer Berlin Heidelberg, 1 edition, 2005. 6

[Kea10] Shuuji Kajita and et al. Biped walking stabilization based on linear inverted pendulum tracking. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010. 6

[Rom24] Bornier Romain. Arm robots controlling by reinforcement learning. https://github.com/R0mB0r/RL_RobotArm, 2024. 26

[SB15] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, second edition, in progress edition, 2015. c© 2014, 2015. 12

[SBM+22] Rohan P. Singh, Mehdi Benallegue, Mitsuharu Morisawa, Rafael Cisneros, and Fumio Kanehiro. Learning bipedal walking on planned footsteps for humanoid robots. In *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, November 2022. 14

[XLY+23] Zichun Xu, Yuntao Li, Xiaohang Yang, Zhiyuan Zhao, Lei Zhuang, and Jingdong Zhao. Open-source reinforcement learning environments implemented in mujoco with franka manipulator, 2023. 16, 18

[ZTM22] Kevin Zakka, Yuval Tassa, and MuJoCo Menagerie Contributors. MuJoCo Menagerie: A collection of high-quality simulation models for MuJoCo, 2022. 18

RAPPORT D'EVALUATION
ASSESSMENT REPORT

Merci de retourner ce rapport par courrier ou par voie électronique en fin du stage à :
At the end of the internship, please return this report via mail or email to:

ENSTA Bretagne – Bureau des stages - 2 rue François Verny - 29806 BREST cedex 9 – FRANCE
☎ 00.33 (0) 2.98.34.87.70 / stages@ensta-bretagne.fr

I - ORGANISME / *HOST ORGANISATION*

NOM / *Name* Tokyo University of Science

Adresse / *Address* 6-3-1, Niijuku, Katsusika-ku, Tokyo 125-8585 Japan

Tél / *Phone (including country and area code)* +81-3-5876-1734

Nom du superviseur / *Name of internship supervisor*
Eiichi YOSHIDA
Fonction / *Function* Professor

Adresse e-mail / *E-mail address* eiichi.yoshida@rs.tus.ac.jp

Nom du stagiaire accueilli / *Name of intern* Romain BORNIER

II - EVALUATION / *ASSESSMENT*

Veuillez attribuer une note, en encerclant la lettre appropriée, pour chacune des caractéristiques suivantes. Cette note devra se situer entre **A (très bien)** et **F (très faible)**
*Please attribute a mark from **A (excellent)** to **F (very weak).***

MISSION / *TASK*

❖ La mission de départ a-t-elle été remplie ? A (B) C D E F
Was the initial contract carried out to your satisfaction?

❖ Manquait-il au stagiaire des connaissances ? ☐ oui/*yes* ☒ non/*no*
Was the intern lacking skills?

Si oui, lesquelles ? / *If so, which skills?*

ESPRIT D'EQUIPE / *TEAM SPIRIT*

❖ Le stagiaire s'est-il bien intégré dans l'organisme d'accueil (disponible, sérieux, s'est adapté au travail en groupe) / *Did the intern easily integrate the host organisation? (flexible, conscientious, adapted to team work)*

A (B) C D E F

Souhaitez-vous nous faire part d'observations ou suggestions ? / *If you wish to comment or make a suggestion, please do so here*

COMPORTEMENT AU TRAVAIL / *BEHAVIOUR TOWARDS WORK*

Le comportement du stagiaire était-il conforme à vos attentes (Ponctuel, ordonné, respectueux, soucieux de participer et d'acquérir de nouvelles connaissances) ?

Did the intern live up to expectations? (*Punctual, methodical, responsive to management instructions, attentive to quality, concerned with acquiring new skills)?*

A (B) C D E F

Souhaitez-vous nous faire part d'observations ou suggestions ? / *If you wish to comment or make a suggestion, please do so here* The student was cooperative and was eager to learn new things, I hope he will keep this attitude.

INITIATIVE – AUTONOMIE / *INITIATIVE – AUTONOMY*

Le stagiaire s'est –il rapidement adapté à de nouvelles situations ? (A) B C D E F
(Proposition de solutions aux problèmes rencontrés, autonomie dans le travail, etc.)

Did the intern adapt well to new situations? (A) B C D E F
(eg. suggested solutions to problems encountered, demonstrated autonomy in his/her job, etc.)

Souhaitez-vous nous faire part d'observations ou suggestions ? / *If you wish to comment or make a suggestion, please do so here* The student was autonomy enough, he tried to acquire new knowledge, especially machine learning this time, and to propose after testing himself.

CULTUREL – COMMUNICATION / *CULTURAL – COMMUNICATION*

Le stagiaire était-il ouvert, d'une manière générale, à la communication ? A (B) C D E F
Was the intern open to listening and expressing himself /herself?

Souhaitez-vous nous faire part d'observations ou suggestions ? / *If you wish to comment or make a suggestion, please do so here* At the beginning he looked a bit reserved but after a while he got used to the life in the laboratory and communicated with other colleagues smoothly.

OPINION GLOBALE / *OVERALL ASSESSMENT*

❖ La valeur technique du stagiaire était : (A) B C D E F
Please evaluate the technical skills of the intern:

III - PARTENARIAT FUTUR / *FUTURE PARTNERSHIP*

❖ Etes-vous prêt à accueillir un autre stagiaire l'an prochain ?

Would you be willing to host another intern next year? [x] oui/*yes* [ ] non/*no*

Fait à _____, le _____
In Tokyo, on September 19, 2024.

Signature Entreprise 吉田英一 Signature stagiaire
Company stamp _____ *Intern's signature*

Merci pour votre coopération
We thank you very much for your cooperation