

# Índice general

<b>1. Introducción</b>	<b>2</b>
1.1. Contexto y motivación . . . . .	2
1.2. Objetivos del trabajo . . . . .	3
1.3. Alcance y limitaciones . . . . .	3
1.4. Metodología . . . . .	5
1.5. Estructura del documento . . . . .	7

# Capítulo 1

## Introducción

### 1.1. Contexto y motivación

Según la página oficial de *Nvidia* [1], una de las mayores impulsoras de esta disciplina, el aprendizaje por refuerzo es una técnica de aprendizaje automático que permite a los robots tomar decisiones basadas en la experiencia. En este trabajo de final de grado se va a estudiar esta doctrina para entender sus conceptos fundamentales y poder crear, mediante la herramienta de *Nvidia*, ‘*IsaacLab*’, distintos entornos capaces de ejecutar este procedimiento.

Las inteligencias artificiales son actualmente una tecnología puntera con una gran cantidad de aplicaciones. Concretamente, el aprendizaje automático se ha aplicado en disciplinas como la medicina (en la generación de reportes de imágenes médicas), las finanzas (en la reserva de órdenes de compra) o la energía (en sistemas de refrigeramiento de bancos de datos) [2]. En la robótica especialmente a tomado un gran protagonismo, donde grandes entidades como *Boston Dynamics* [3], han empezado a implementar esta técnica en tareas de locomoción [3].

El interés en este proyecto nace de la idea de aplicar esta herramienta dentro del proyecto *ROMERIN*, un robot modular escalador para la inspección de infraestructuras [4]. Rápidamente se vio, debido a la compleja naturaleza de esta herramienta, que era necesario realizar un estudio completo. Añadido a la generosidad del grupo trabajando en el proyecto *MetaTool* a la hora de formar, se decidió colaborar estrechamente dentro de este proyecto, analizando y revisando código.

## 1.2. Objetivos del trabajo

En este proyecto se busca realizar un estudio del aprendizaje por refuerzo y la herramienta *IsaacLab* con dos objetivos principales. En primer lugar, realizar labores dentro del proyecto europeo *MetaTool* y, en segundo lugar, dar una visión global de la herramienta para que futuros estudiantes puedan basarse en ella para realizar su propio aporte a esta disciplina.

Para alcanzar estos objetivos, se proponen una serie de objetivos secundarios:

1. Obtener una visión general del impacto del aprendizaje por refuerzo en el campo de la robótica.
2. Estudiar las bases del aprendizaje por refuerzo, centrándose en su estructura y sus principales algoritmos.
3. Explicar el funcionamiento de la herramienta *IsaacLab* para su aplicación en aprendizaje por refuerzo.
4. Analizar un ejemplo de dicha herramienta para profundizar en ella y proveer de una guía práctica para trabajos futuros.
5. Estudio del proyecto *MetaTool*: Misión y Visión.
6. Realización de trabajos prácticos en el proyecto con la herramienta *IsaacLab*
7. Estudio del problema *Sim2Real* y posibles soluciones
8. Realización de un código para la implementación de políticas.

Cumplidos todos estos objetivos se espera poder crear una base fuerte para comenzar a implementar esta herramienta dentro de la ETSIDI, con el objetivo de implementarse en futuros proyectos.

## 1.3. Alcance y limitaciones

En este TFG cubriremos el proceso para realizar entornos de aprendizaje automático. Al tener este objetivo en mente en este TFG se podrán encontrar distintos aspectos y fases de esta disciplina. Esto contempla desde los aspectos más fundamentales de la teoría del aprendizaje automático, hasta las distintas estructuras de datos, clases y ficheros que ejecutan y simulan los entornos. Primero de todo, para situarse dentro del marco del Aprendizaje por Refuerzo en robótica se revisará el estado actual del arte. Se presentarán los avances más importantes en distintos campos de la robótica, entre ellos la

manipulación, la locomoción y otras aplicaciones como drones, navegación o dispositivos de visión.

A continuación, se explicará la teoría fundamental del Aprendizaje por Refuerzo. En una primera instancia, se presentará la estructura principal que se utiliza en esta disciplina y sus partes, entre las que se encuentran los agentes, los entornos y sus interacciones (acciones, observaciones y recompensas). Dentro de este marco teórico se estudiará los procesos de decisión Márkov (MDP), en los cuales se asienta la base de los algoritmos que se utilizarán. Una vez estudiado esto, se presentarán algunos de estos algoritmos, desde los más simples (*Monte Carlo*, *TD*) hasta los que utilizaremos en las simulaciones (*PPO*, *SAC*, *A2C*). Por último, se estudiarán las distintas métricas de evaluación.

Una vez desarrollado el marco teórico, se procederá a introducir la herramienta *IsaacLab*. Después de una primera introducción a la herramienta, se comenzará a explicar sus distintas funcionalidades. Primero, se explicará qué es y cómo crear una escena. Una vez definidas las escenas, se desarrollará las dos principales arquitecturas de los entornos, la manera directa y la basada en manejadores. Dentro de estas arquitecturas se entrará en detalle sobre sus principales funciones y clases. Definidas las arquitecturas, se expandirá en sus estructuras de datos relacionadas con tensores, realizando un breve estudio de la herramienta *pyTorch*, estudiándose su funcionamiento y sus principales métodos. Seguidamente se estudiarán los principales scripts de entrenamiento para analizar este proceso, realizando el mismo proceso con los scripts de evaluación. Por último, se presentará una idea general de la herramienta en su conjunto.

En el punto número cinco se estudiará un caso ejemplo de la herramienta *IsaacLab*. Primero, se presentará el ejemplo escogido y la motivación detrás de esta elección. Seguidamente, se presentará la estructura de ficheros y directorios donde se agrupan las distintas partes del código. Una vez presentada dicha estructura, se estudiará cada fichero por separado, analizando su aporte a la ejecución global de la simulación del entorno. Una vez definida la función de cada fichero, se presentará el flujo de ejecución en el entrenamiento de la red neuronal. Por último, se estudiarán los resultados obtenidos y se presentarán posibles mejoras al problema.

Habiendo estudiado las distintas características del RL, se comenzará a intervenir en el proyecto MetaTool. En este apartado, definiremos el contexto del proyecto *MetaTool* y cuál es su principal objetivo. Seguidamente, se concretará las tareas en las cuales se intervendrá y el objetivo de la participación. Una vez definido el trabajo a realizar, se realizará un estudio del caso concreto en el que se desarrollará la actividad. Después, se realizará una depuración del código realizando las correcciones oportunas para mejorar su funcionamiento. Por último, se realizará una evaluación del entorno en su conjunto y sus posibles limitaciones.

Otro punto importante del Aprendizaje por Refuerzo que se estudiará en este trabajo es el problema del Sim2Real, que consiste en la aplicación de las redes neuronales entrenadas para el control en el entorno real. Primero, se presentará el concepto de Sim2Real y sus principales desafíos. Seguidamente, se enumerarán y analizarán las distintas técnicas para realizar este trasvase a la realidad. Posteriormente, se estudiará un caso de ejemplo en el que se realiza este ejercicio, aplicado a un robot *Kinova*. Con los conocimientos obtenidos en este estudio, se realizará un ejercicio propio con el caso MetaTool anteriormente estudiado, adaptándolo al robot utilizado (*UR5*). Finalmente se evaluará el enfoque utilizado y los resultados obtenidos.

En la última parte del trabajo, se expondrán las conclusiones al proyecto en su conjunto, valorando las aportaciones realizadas, las dificultades encontradas y las oportunidades de trabajos futuros. En este trabajo, debido a la gran extensión de esta disciplina, se dejan de cubrir algunos paradigmas. Por un lado, únicamente se estudia dentro del aprendizaje automático el aprendizaje por refuerzo, dejando fuera el aprendizaje supervisado y no supervisado. Además, pese a que se realice aprendizaje por refuerzo profundo, no se entrará en detalle en el aprendizaje profundo. Se estudiarán las características de las redes neuronales, pero no se profundizará en la matemática detrás de ellas, para centrarnos así en la creación de entornos para este propósito.

## 1.4. Metodología

En este trabajo de final de grado existen principalmente dos líneas, una parte teórica acerca del aprendizaje por refuerzo y una parte práctica mediante programación en Python y finalmente URsim. Por otro lado, la preparación de este documento se ha realizado después de 5 meses realizando tareas de programación e investigación por propia cuenta o en conjunto con el equipo de investigación del proyecto MetaTool. A continuación, se expondrá la metodología característica de cada apartado.

La introducción, en primer lugar, se ha preparado después de haber realizado la mayoría de las labores teóricas y prácticas del trabajo. Teniendo así una visión general del trabajo global, se han expuesto las distintas características del trabajo y su enfoque general.

Para el estado del arte, al querer mostrar una visión general del estado del aprendizaje por refuerzo en la robótica, se han buscado distintos artículos de investigación sobre esta disciplina y sus aplicaciones prácticas. Al haber realizado este ejercicio después de este periodo de aprendizaje y práctica, se han podido identificar los factores más importantes de cada artículo, así como identificar los artículos más relevantes para este ejercicio.

En cuanto al apartado 3, en el cual se exponen los fundamentos teóricos del Apre-

dizaje por Refuerzo, se ha seguido la siguiente metodología. En primer lugar, se estudio un curso de aprendizaje por Refuerzo impartido por David Silver [5]. Mediante este curso se obtuvo una visión general de esta disciplina, entendiendo su estructura general y la base para sus principales algoritmos. Una vez obtenida una visión general, y después de aplicar esta visión en labores prácticas, se estudió más específicamente cada elemento, indagando en distintas fuentes de información.

El 4 apartado, acerca de la herramienta *IsaacLab*, se trató de manera distinta. Al ser *IsaacLab* una herramienta concreta y propiedad de una entidad privada, Nvidia, existe menos diversidad de información acerca de ella. Por tanto, para su estudio, se utilizó principalmente la información contenida en sus fuentes oficiales. En este apartado concreto, se utilizó principalmente los tutoriales proporcionados por la plataforma para el aprendizaje de su estructura y aplicación, así como los distintos glosarios de las funciones y clases; y su información detallada acerca de la propia plataforma y sus bibliotecas. A esto se le sumo el conocimiento aprendido en el trabajo en conjunto con el equipo MetaTool, en especial con Virgilio Gómez, especialista de la plataforma y líder de la división en la que se trabajó.

El apartado 5, al constar de un análisis concreto de un ejemplo proporcionado por la plataforma, se ha utilizado los conocimientos aprendidos en el apartado anterior. Para realizar este análisis se ha seguido el siguiente proceso. En primer lugar, se analizó todo el ejercicio en su conjunto, estudiando los distintos ficheros que se ponen en ejecución. Con esta estructura identificada, se realizó un diagrama de clases y secuencial, identificando las distintas llamadas y las funciones utilizadas. Con este diagrama en mente, se estudió cada fichero por si solo, explicando la función de cada apartado del código y su aportación global al ejercicio.

A su vez el apartado 6 se tratará de una forma distinta, pues en vez de ser un ejercicio ejemplo propio de la plataforma, es un caso práctico del proyecto MetaTool. Por ello, antes de analizarse el ejercicio se realizará un pequeño estudio del proyecto general. Con el enfoque general en mente, se realizará un estudio propio del caso en cuestión. A diferencia del apartado anterior, este estudio se basará en la depuración realizada del código, en el cual se realizaron una serie de correcciones para su correcto funcionamiento. El apartado 7, enfocado a la implementación en el robot del ejercicio anterior, se estudió un caso real de esta implementación, pero en un robot distinto. Estudiado este caso, se adaptó para su uso en el *UR5 Robohabilis*, utilizando el lenguaje *URSim* para el control del Robot.

Por último, una vez terminado el trabajo, se expondrá las conclusiones obtenidas de este, basándose plenamente en la experiencia obtenida en el transcurso del proyecto.

En conclusión, este trabajo sigue distintas metodologías, dependiendo si se trata de

un enfoque práctico o teórico. Tomando en conjunto todo, se podría definir una metodología general. Primero, se realiza el estudio teórico, tanto del aprendizaje en refuerzo general como el de la herramienta específica. Después, se utiliza los conocimientos obtenidos para realizar estudios prácticos y analíticos.

## **1.5. Estructura del documento**

PENDIENTE