# Borrador Memoria TFG 2.0

***Entorno de simulación para el entrenamiento mediante Reinforcement Learning del vuelo autónomo de un cuadricóptero***

**Índice**

1. *Introducción*
2. *Objetivos*
3. *Resultados*
4. *Conclusiones*
5. *Bibliografía*

*Anexos:*

* *Estado del Arte*

## INTRODUCCION

Presentar los drones. Contextualizar su uso hoy en dia (y el que se dará mañana), e introducir la necesidad de tener sistemas para entrenar drones inteligentes.

Introducir el RL y explicar la necesidad de hacer pruebas “una y otra vez”

Una de las tecnologías que más destacan y llaman la atención en estos últimos años son los llamados drones, definidos formalmente como *aeronaves no tripuladas* por la RAE [http://dle.rae.es/?id=ED2QqnQ]. Estos aparatos se han convertido en una atracción para casi cualquier ámbito, comenzando en usos militares y llegando a ser juguetes de ocio, no sin pasar por usos comerciales como el transporte o relacionados con seguridad y defensa.

Su bajo coste, reducido tamaño y la posibilidad de manejarlos remotamente les da un potencial inmenso para una infinidad de usos, entre otras cosas, para tareas que debería hacer una persona pudiendo tener que asumir riesgos o peligros, como puede ser la vigilancia desde el aire en zonas de conflicto, la búsqueda de supervivientes tras una catástrofe natural o la actuación en incendios forestales.

Sin embargo, para algunas tareas los recursos humanos son limitados y no es viable ocupar a una persona pilotando un dron, apareciendo de esta forma la necesidad de diseñar aeronaves que puedan llevar a cabo su cometido de forma autónoma.

Volar de forma autónoma es una tarea muy compleja en la que pueden suceder innumerables situaciones diferentes e imprevisibles, a las que se debe dar respuesta de la forma más rápida y acertada posible. Es necesario dotar a los drones de inteligencia para que puedan desenvolverse por sí mismos y aprender de su propia experiencia.

Y esta necesidad pide a gritos una solución de RL: el RL es un área del Machine Learning que agrupa técnicas de aprendizaje automático basadas en la experimentación y la experiencia de la maquina. Estas técnicas comparten la necesidad de un entrenamiento de prueba y error, en el que se deben realizar una serie de intentos de llegar a la meta, durante los cuales el aparato va aprendiendo lo que debe hacer y lo que no para llegar a su objetivo. Es decir, hay que repetir una y otra vez la misma situación a resolver, y se deja que el dron interactue y descubra el entorno que lo rodea.

En este trabajo fin de grado se pretende crear un freamwork de simulación, desarrollo y entrenamiento de sistemas RL orientado a su uso en/para drones.

En este trabajo fin de grado se pretende experimentar con el uso de un entorno de simulación [ Referencia a Anexo Simuladores ] para entrenar un cuadricóptero con técnicas basadas en *Reinforcement Learning* [ Referencia a Anexo RL ] y conseguir que este sea capaz de volar de forma autónoma en la realidad.

## OBJETIVOS

Objetivo principal: crear un framework de simulación, desarrollo y testeo de sistemas RL realista, versátil y plug&play.

* El objetivo de este tfg es crear un framework de simulación y entrenamiento para drones. Es decir, un conjunto de herramientas que nos ayuden a diseñar y probar algoritmos, así como a entrenar el dron con esos algoritmos. De forma que una vez el dron esté preparado, solo haya que mover la lógica (incluyendo ese “conocimiento generado”) al dron real, y a volar.
* Ventajas de esto? Que no hay que modificar nada, lo que se entrena es lo mismo que va a llevar el dron real (filosofia plug&play)

El objetivo final de este trabajo es el desarrollo de un freamwork de simulación, desarrollo y entrenamiento de sistemas RL, es decir, un conjunto de herramientas que faciliten el desarrollo y posterior entrenamiento de sistemas inteligentes (basados en RL) para drones, siguiendo una filosofía plug&play de cara al desarrollador.

La ventaja principal de este framework reside en que todas las implementaciones son migradas de la simulación al equipo real de forma casi directa, sin necesidad de hacer ningún cambio en la implementación. Esto se consigue gracias a la integración en la simulación de un firmware de autopiloto creado para equipos reales, por lo que el sistema desarrollado se programará para la simualción exactamente igual que se haría para el dron físico.

## RESULTADOS

* Explicar los bloques que componen el sistema completo (airsim, px4, ros, gym) y cómo se integran.
* A continuación, explicar las implementaciones llevadas a cabo en cada uno de los bloques. (airsim con integración de ROS, px4 modificado para poder hacer el reset, ROS + GYM para unir todo y hacer el sistema de entrenamiento)
* Por último, mencionar las pruebas y el desarrollo en general que se llevó a cabo durante la implementación de esos módulos.

## CONCLUSIONES

Finalmente que hemos conseguido? Hemos creado un framework con X ventajas y Y inconvenientes. Soluciona estos problemas del estado del arte.

Recalcar que se ha creado algo nuevo que aporta valor, y que no existía previamente.

## ANEXOS

### ESTADO DEL ARTE

Dado que el objetivo es tener un framework de simulación, desarrollo y testeo de sistemas RL, podemos hablar de los diferentes frameworks que hay para simulación (aptdo simuladores).

Dentro de los frameworks hay que mencionar al final de todo, cuando ya nos centramos en airsim, que existe un freamwork similar al que hemos creado pero que utiliza el simple\_flight. La ventaja del nuestro esta en que utiliza el px4, ya que en las ultimas versiones de airsim, integra ros tb, cosa queno hacia al principio.

Acabar hablando de por que escogemos airsim? Si no, se puede meter en SIMULADOR, pero pienso que quedará mejor aquí.

### TECNOLOGÍAS EMPLEADAS

* SIMULADOR: centrarse en el airsim, hablar de sus ventajas y desventajas, compatibilidades e incompatibilidades.
* AUTOPILOTO: explicar qué es y para que sirve, cuál se ha utilizado y por qué.
  + HITL vs SITL: hablar también de la existencia de dos modos: HITL y SITL, y de cuál se ha utilizado y por qué.
* ROS: explicar qué es y por qué se ha utilizado.
* RL: explicar por encima lo que es y en qué se basa, por qué es útil en nuestro contexto, y cómo funciona el entrenamiento (hacer ver la necesidad de una función de reset).