

# Simulación de un Agente Robótico Scara que aprende a jugar AIR HOCKEY mediante Twin Delayed DDPG (TD3)

Patrichs Alexander Inocente Valle<sup>1</sup>

<sup>1</sup>Universidad Nacional de Ingeniería. Facultad de Ciencias.  
Ciencia de la Computación

Enero, 2023



**FACULTAD DE  
CIENCIAS**

# Tabla de contenidos

- 1 Introducción
- 2 Problema
- 3 Objetivo
- 4 Conocimientos previos
- 5 Herramientas Utilizadas
- 6 Análisis del Entorno
- 7 Metodología y desarrollo
- 8 Entrenamiento
- 9 Resultados
- 10 Conclusiones
- 11 Trabajos futuros

# Contenido

1 Introducción

2 Problema

3 Objetivo

4 Conocimientos previos

5 Herramientas Utilizadas

6 Análisis del Entorno

7 Metodología y desarrollo

8 Entrenamiento

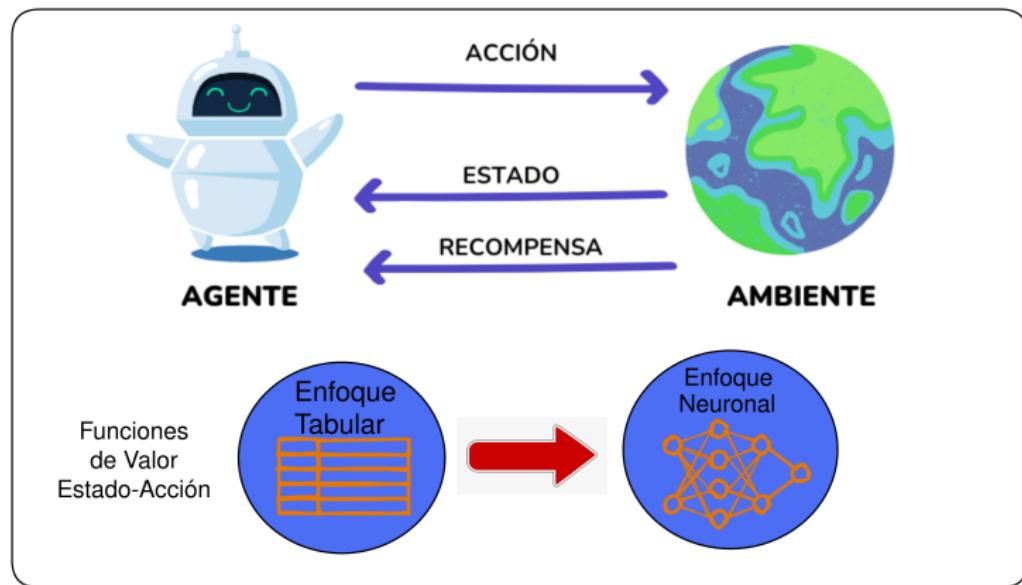
9 Resultados

10 Conclusiones

11 Trabajos futuros

# Introducción

Dentro del campo del Aprendizaje por Refuerzo, inicialmente un enfoque tabular para asignar el puntaje de calidad a un par (estado,acción) dado era suficiente, sin embargo aquellos problemas que incorporaban estados de dimensiones altas tuvieron que migrar a un enfoque de Red Neuronal.



# Contenido

- 1 Introducción
- 2 Problema
- 3 Objetivo
- 4 Conocimientos previos
- 5 Herramientas Utilizadas
- 6 Análisis del Entorno
- 7 Metodología y desarrollo
- 8 Entrenamiento
- 9 Resultados
- 10 Conclusiones
- 11 Trabajos futuros

# Problema

Tenemos por lo tanto métodos que nos permiten trabajar con acciones discretas resolviendo entornos tal como pacman, pong de atari y otros similares, pero, **si deseamos trabajar con señales continuas**, para controlar por ejemplo las acciones de un robot (ángulo de giro de articulaciones o velocidades) **¿Qué métodos de aprendizaje por refuerzo usamos?** debemos dar una mirada a otras alternativas:

- Binning: discretizamos una acción continua en intervalos, volviendo un problema de acciones continuas en un problema de acciones discretas y aplicamos los mismos métodos.
- Métodos Actor - Crítico: Trabajan con acciones continuas haciendo uso de lo mejor de dos mundos: Q-Learning (Quien Criticará) y política de gradientes (Quién actuará). En esta investigación vamos a centrarnos en un método Actor-Crítico: El metodo TD3, el cual es una mejora del método DDPG.

# Contenido

- 1 Introducción
- 2 Problema
- 3 Objetivo
- 4 Conocimientos previos
- 5 Herramientas Utilizadas
- 6 Análisis del Entorno
- 7 Metodología y desarrollo
- 8 Entrenamiento
- 9 Resultados
- 10 Conclusiones
- 11 Trabajos futuros

# Objetivo

El principal objetivo del presente trabajo de investigación es desarrollar un agente Robótico Scara que pueda aprender a jugar Air Hockey, con la motivación de incursionar mas a profundidad en el área de Aprendizaje por Refuerzo de acciones Continuas.

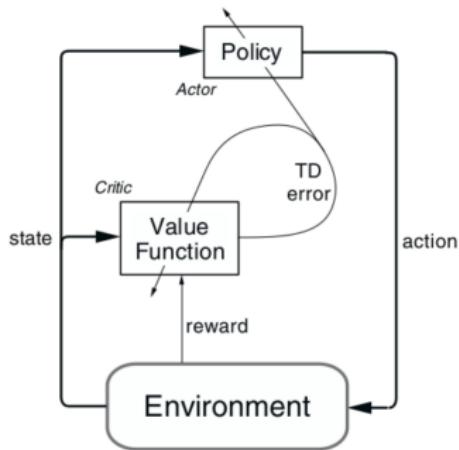


**Figura:** Demostración de juego Air Hockey.

# Contenido

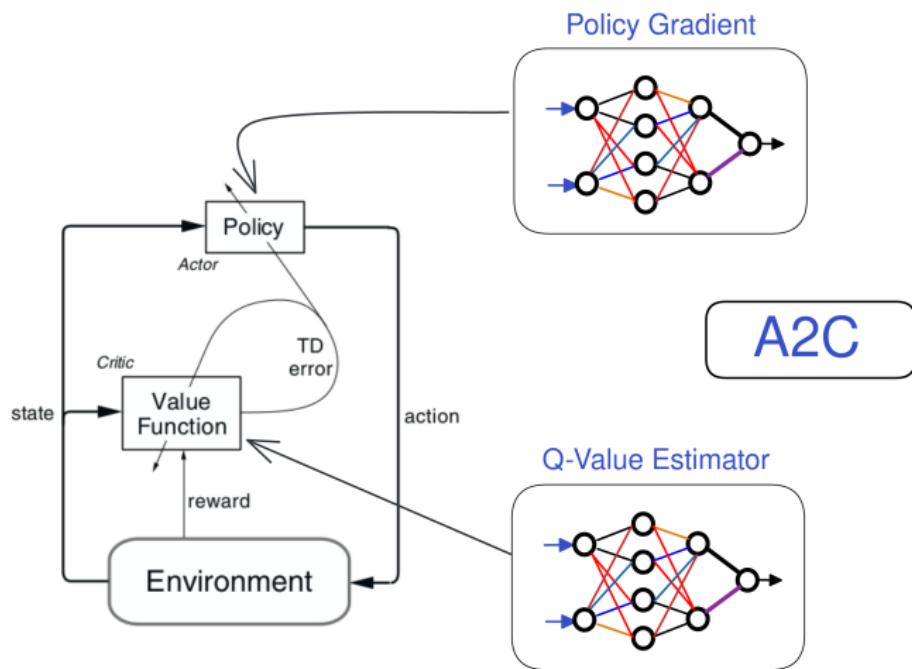
- 1 Introducción
- 2 Problema
- 3 Objetivo
- 4 Conocimientos previos
- 5 Herramientas Utilizadas
- 6 Análisis del Entorno
- 7 Metodología y desarrollo
- 8 Entrenamiento
- 9 Resultados
- 10 Conclusiones
- 11 Trabajos futuros

# Modelo de Actor Crítico

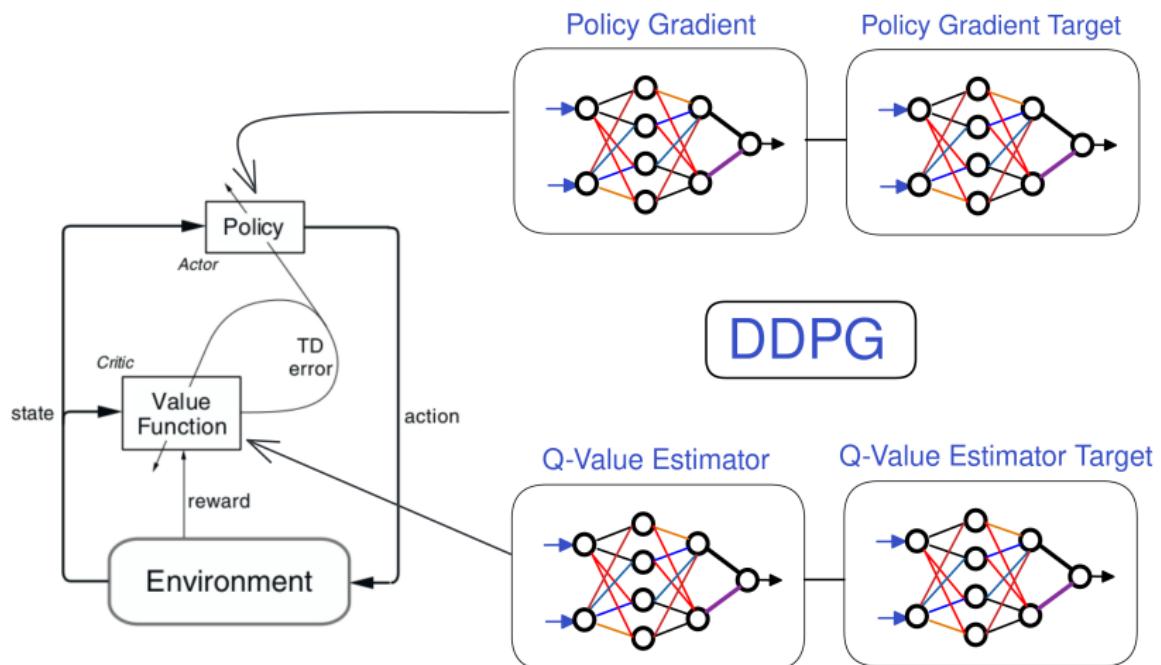


- ① El actor tomará una acción.
- ② El Crítico evaluará la acción tomada, asignando un puntaje.
- ③ El Crítico mejorará disminuyendo el TD error.
- ④ El Actor mejorará buscando que sus acciones sean de mayor puntaje en la evaluación del crítico.

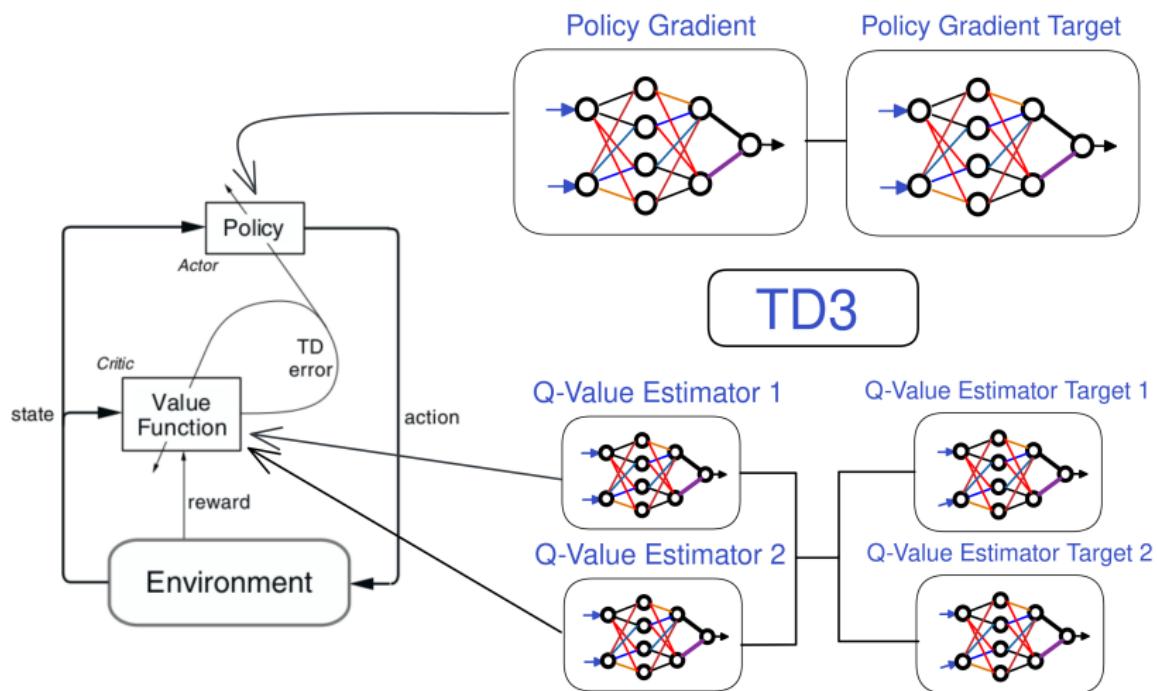
# Modelo de Actor Crítico



# Modelo de Actor Crítico - DDPG



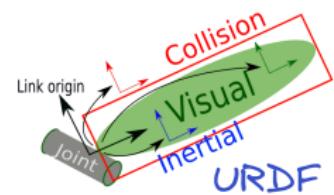
# Modelo de Actor Crítico - TD3



# Contenido

- 1 Introducción
- 2 Problema
- 3 Objetivo
- 4 Conocimientos previos
- 5 Herramientas Utilizadas
- 6 Análisis del Entorno
- 7 Metodología y desarrollo
- 8 Entrenamiento
- 9 Resultados
- 10 Conclusiones
- 11 Trabajos futuros

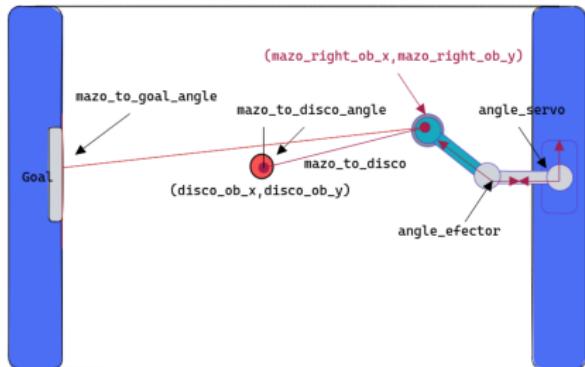
# Herramientas Utilizadas



# Contenido

- 1 Introducción
- 2 Problema
- 3 Objetivo
- 4 Conocimientos previos
- 5 Herramientas Utilizadas
- 6 Análisis del Entorno
- 7 Metodología y desarrollo
- 8 Entrenamiento
- 9 Resultados
- 10 Conclusiones
- 11 Trabajos futuros

# Análisis del Entorno

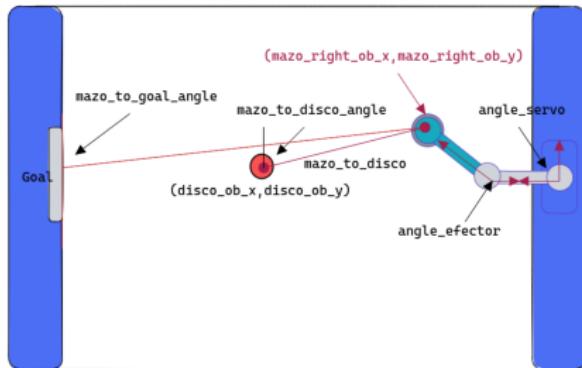


Modo de Control	Elemento	Parámetro	Unidades	Rango
POSITION CONTROL	Unión 1	targetPosition	Radian	[-2.0,2.0]
POSITION CONTROL	Unión 1	maxVelocity	m/s	[0,10]
POSITION CONTROL	Unión 2	targetPosition	Radian	[-2.9,2.9]
POSITION CONTROL	Unión 2	maxVelocity	m/s	[0,10]

## Espacio de Observación

Variable	Descripción	Unidades	Rango
disco_ob_x	Posición x de Disco	m	[−5.07,5.07]
disco_ob_y	Posición y de Disco	m	[−7.49,7.49]
mazo_right_ob_x	Posición x de Mazo	m	[−4.99,4.99]
mazo_right_ob_y	Posición y de Mazo	m	[3.15,7.42]
angle_servo	Ángulo de Giro de 1era Unión	Radian	[−2.0,2.0]
angle_efector	Ángulo de Giro de 2da Unión	Radian	[−2.9,2.9]
mazo_to_disco	distancia entre mazo y disco	m	[0.85,18.0]
mazo_to_disco_angle	Ángulo entre disco y mazo	Radian	[−3.1415,3.1415]
mazo_to_goal_angle	Ángulo entre portería y mazo	Radian	[1.25,1.89]
disco_to_goal_angle	Ángulo entre portería y disco	Radian	[0,3.15]

# Análisis del Entorno



Modo de Control	Elemento	Parámetro	Unidades	Rango
POSITION CONTROL	Unión 1	targetPosition	Radian	[-2.0,2.0]
POSITION CONTROL	Unión 1	maxVelocity	m/s	[0,10]
POSITION CONTROL	Unión 2	targetPosition	Radian	[-2.9,2.9]
POSITION CONTROL	Unión 2	maxVelocity	m/s	[0,10]

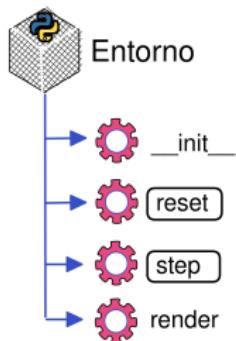
## Función de Recompensa

Evento	Recompensa	Tipo	Descripción
Mazo Golpea disco.	+50	No Terminal	Estimula al agente a golpear disco.
Mazo delante de Disco.	-0.02	No Terminal	Estimula al agente a poder retroceder para defenderse.
Disco Anota en la portería contraria.	+100	Terminal	Estimula a lograr el objetivo final.
Disco entra en la portería del propio agente.	-100	Terminal	Estimula a moverse para defender su propia portería.
Disco golpea linea enemiga.	(507-abs(disco._ob_x)*100)	Terminal	Estimula al agente a apuntar mejor hacia el centro de linea enemiga.
Mazo choca contra pared.	-1.0	No Terminal	Evita que el robot se dañe.

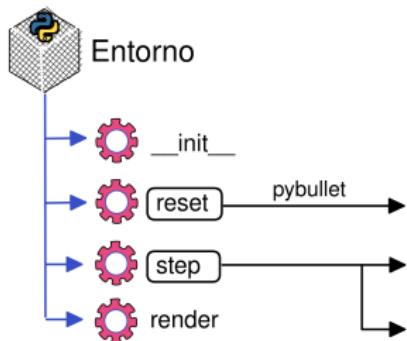
# Contenido

- 1 Introducción
- 2 Problema
- 3 Objetivo
- 4 Conocimientos previos
- 5 Herramientas Utilizadas
- 6 Análisis del Entorno
- 7 Metodología y desarrollo
- 8 Entrenamiento
- 9 Resultados
- 10 Conclusiones
- 11 Trabajos futuros

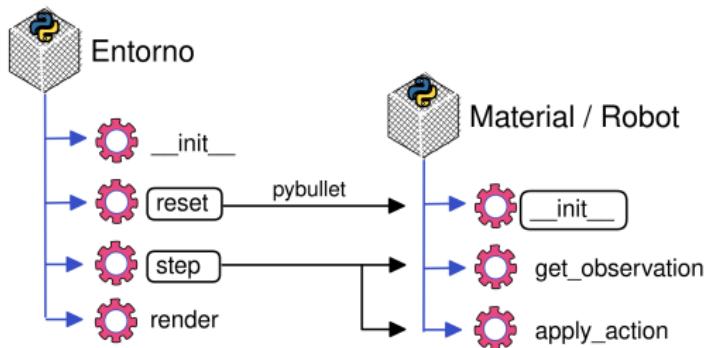
# Implementación de Entorno



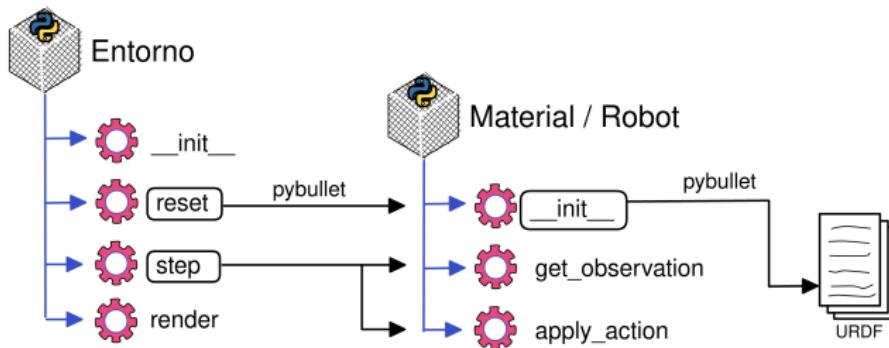
# Implementación de Entorno



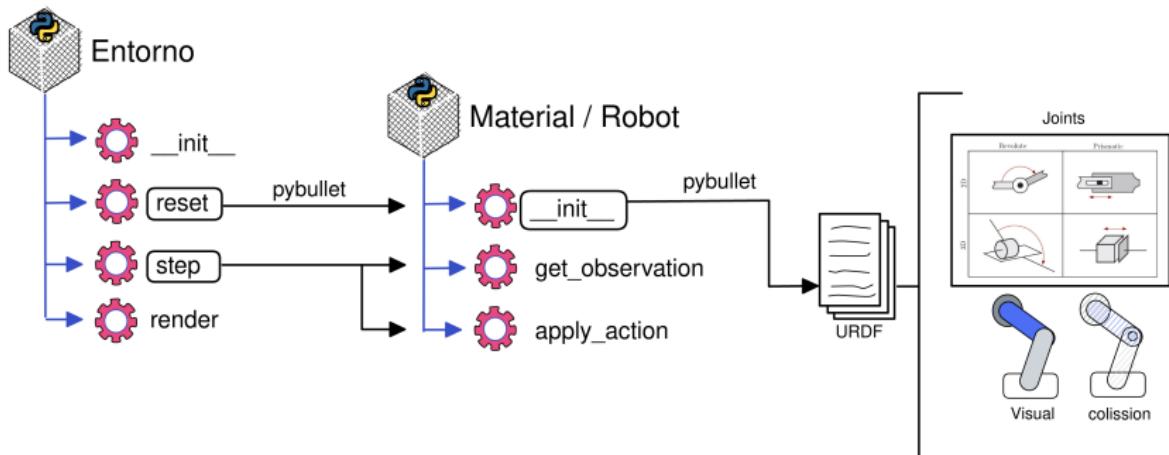
# Implementación de Entorno



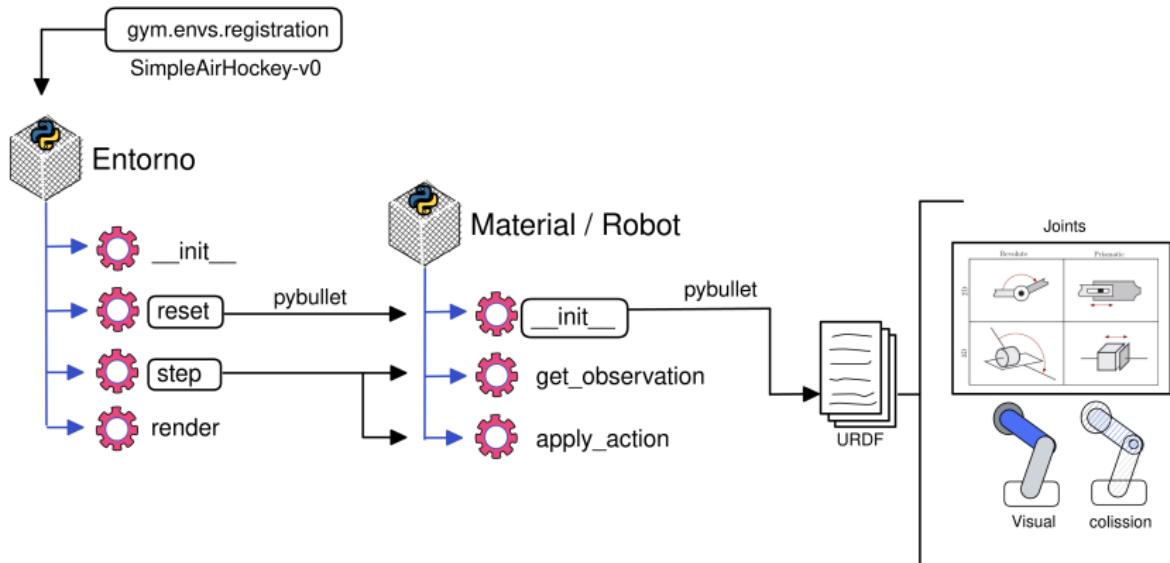
# Implementación de Entorno



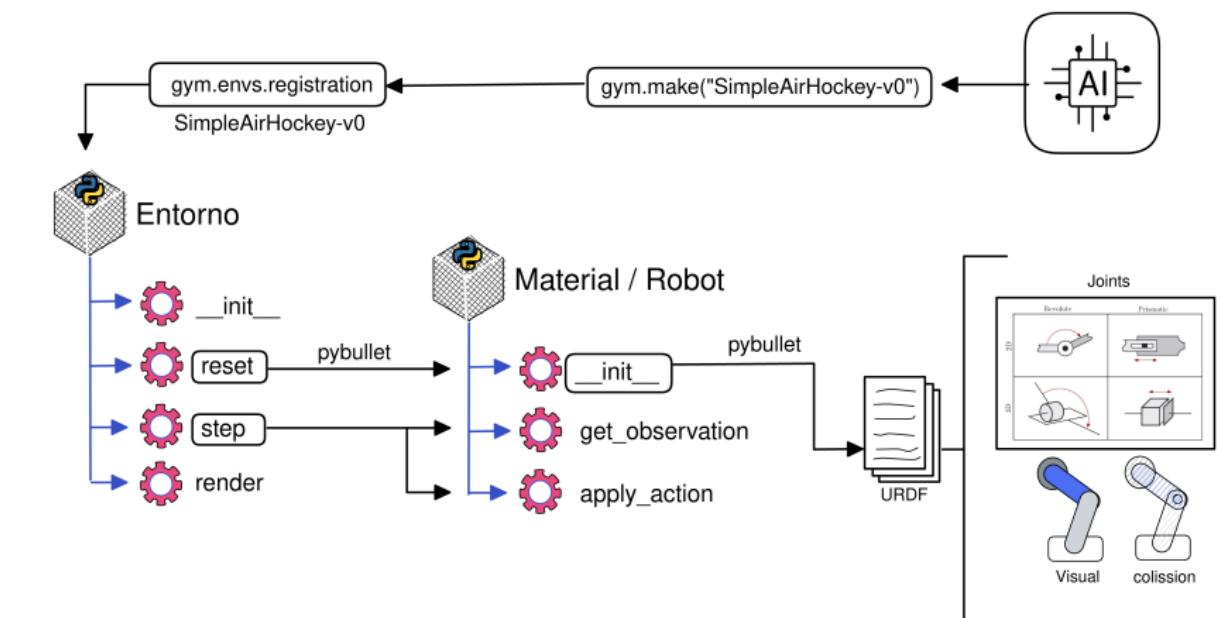
# Implementación de Entorno



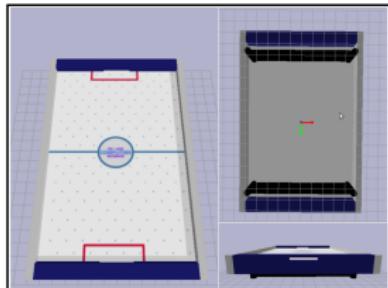
# Implementación de Entorno



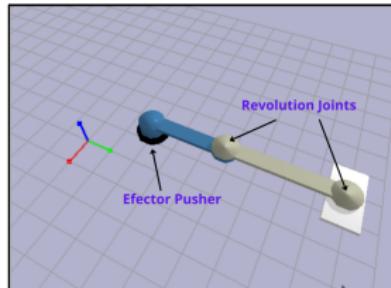
# Implementación de Entorno



# URDF - Materiales



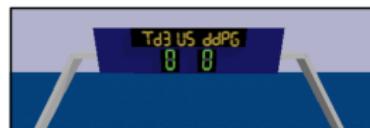
Tablero Air Hockey



Robot Scara

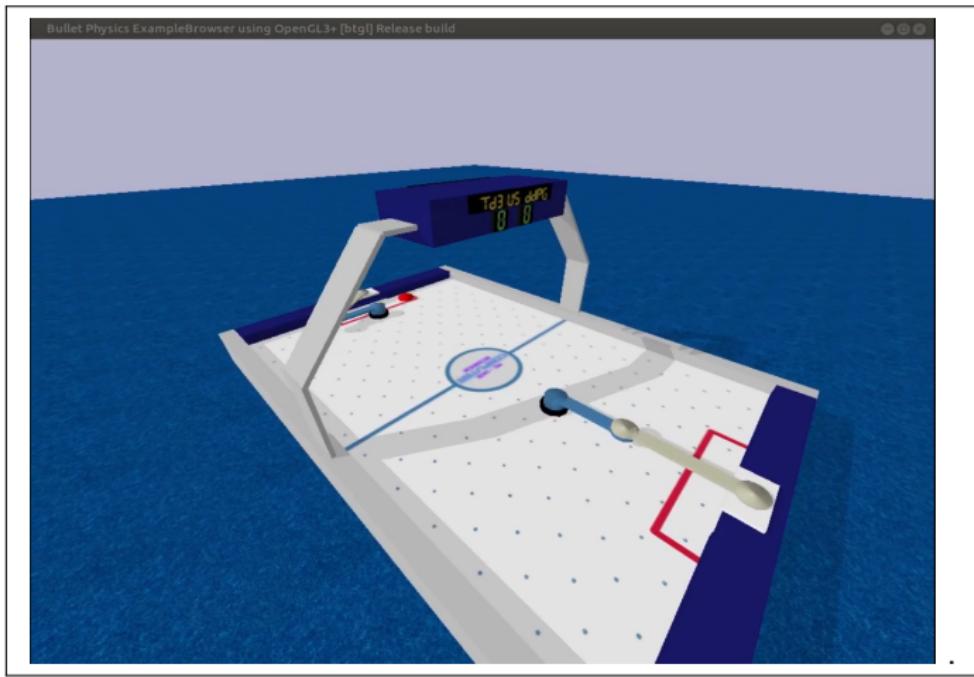


Disco



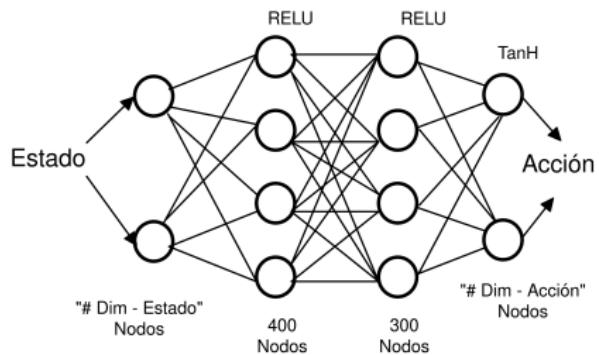
Marcador de Puntaje

# Uniendo Las Piezas

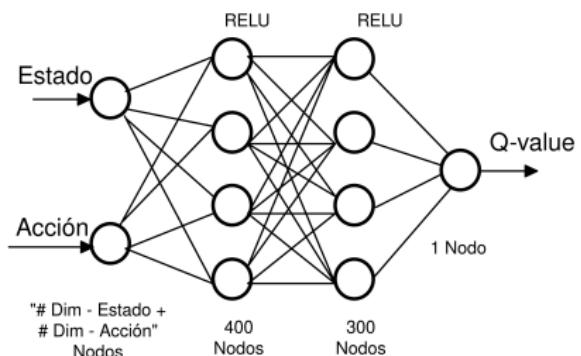


# Implementación de TD3 - Arquitecturas

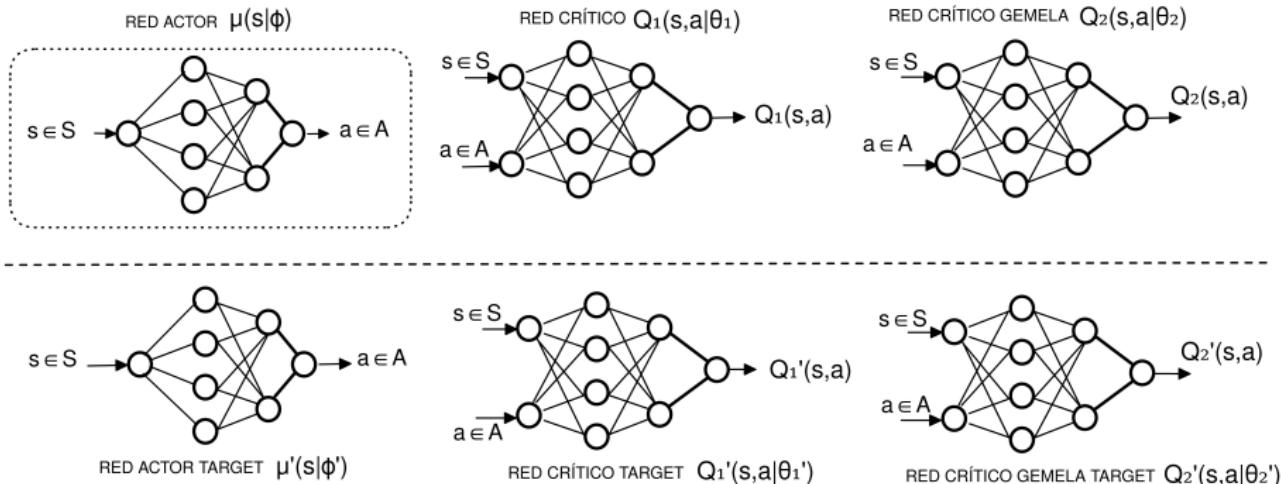
ARQUITECTURA ACTOR



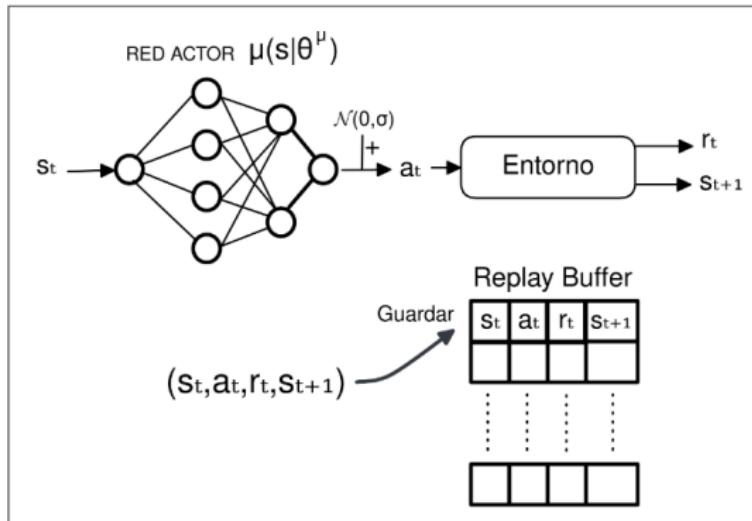
ARQUITECTURA CRÍTICO



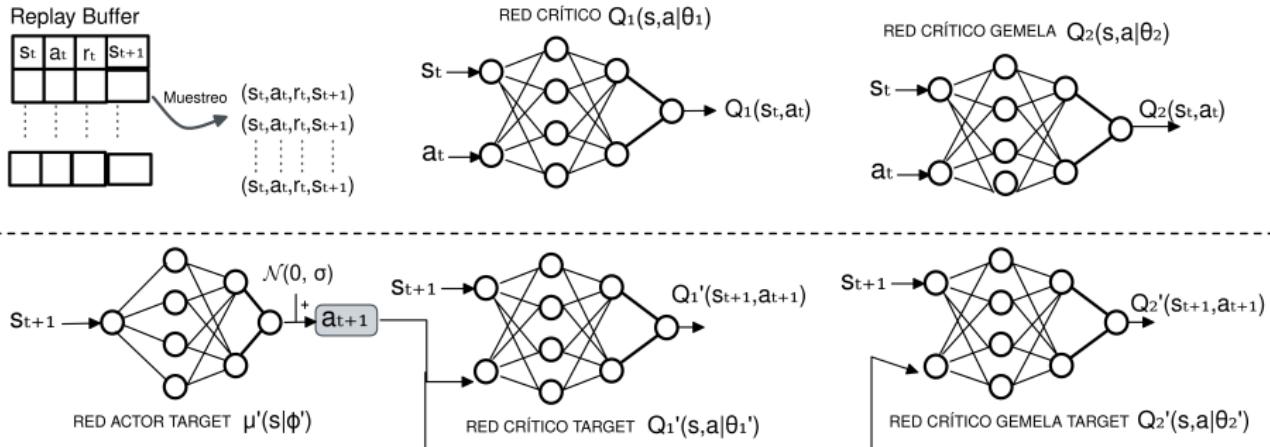
# Implementación de TD3 - Modelo Completo



# Implementación de TD3 - Paso 1



# Implementación de TD3 - Paso 2



# Implementación de TD3 - Paso 3

$$\delta_{t1} = R(s_t, a_t) + \gamma(\min(Q'_1(s_{t+1}, a_{t+1}), Q'_2(s_{t+1}, a_{t+1}))) - Q_1(s_t, a_t)$$

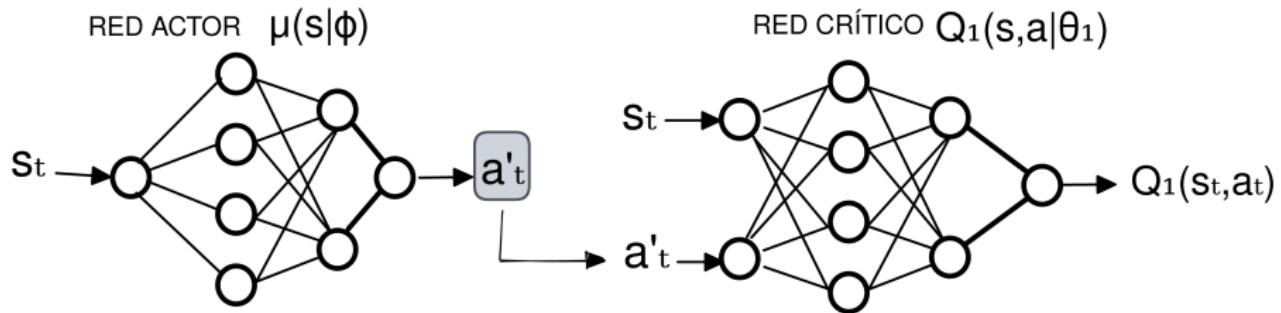
$$\delta_{t2} = \underbrace{R(s_t, a_t) + \gamma(\min(Q'_1(s_{t+1}, a_{t+1}), Q'_2(s_{t+1}, a_{t+1}))) - Q_2(s_t, a_t)}$$

Por la Ecuación de Bellman

$$\begin{aligned} Loss &= Loss(Q_1) + Loss(Q_2) \\ &= MSE(\delta_{t1}) + MSE(\delta_{t2}) \\ &= \frac{1}{N} (\sum (\delta_{t1})^2 + \sum (\delta_{t2})^2) \end{aligned}$$

# Implementación de TD3 - Paso 4

Cada Dos pasos se actualiza la Red Actor.



$$J = \frac{1}{N} \left( \sum Q(s_t, a'_t) \right), \text{ donde } a'_t = \mu(s_t | \theta^\mu)$$

$$\nabla J = \frac{1}{N} \sum (\nabla_{a'_t} Q(s_t, a'_t) \nabla_{\theta^\mu} \mu(s_t | \theta^\mu))$$

# Implementación de TD3 - Paso 5

Las Redes Target se actualizan cada dos pasos mediante promedio Poliak, tal como se muestra a continuación:

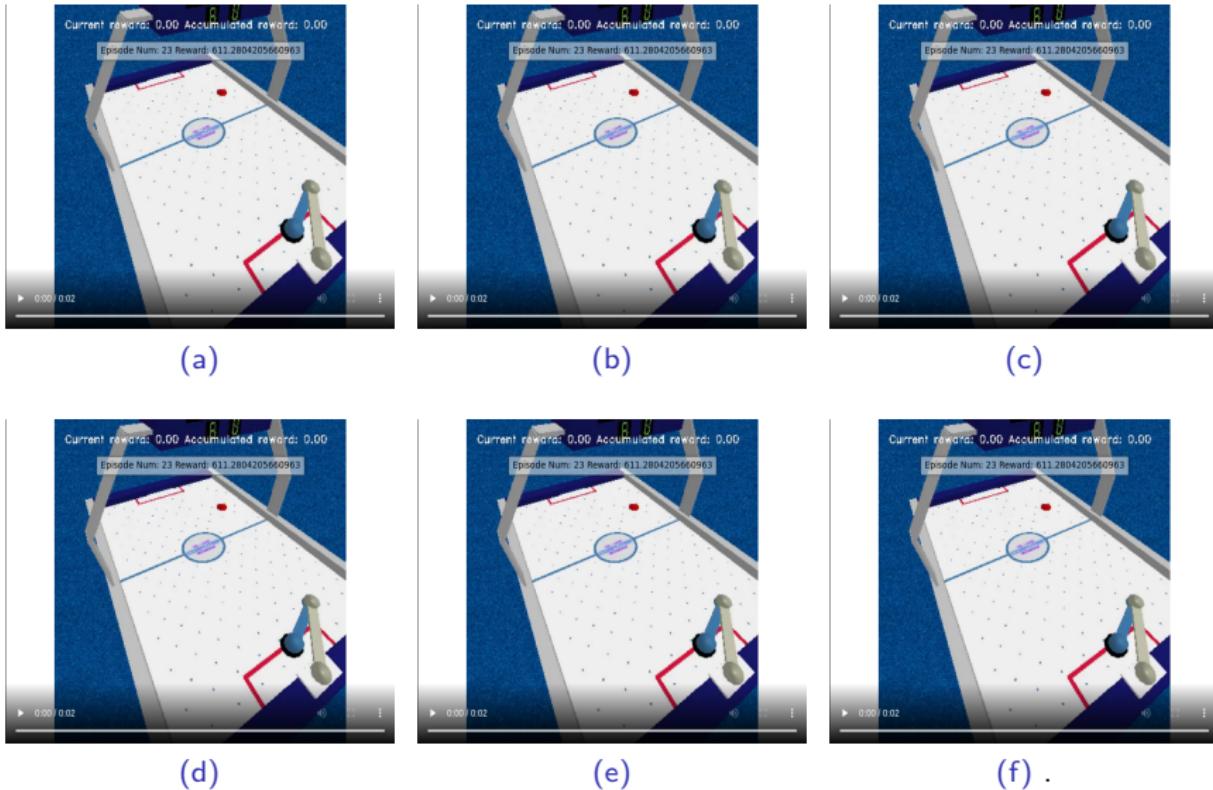
$$\theta^{Q'} \leftarrow \tau\theta^Q + (1 - \tau)\theta^{Q'}$$

$$\theta^{\mu'} \leftarrow \tau\theta^\mu + (1 - \tau)\theta^{\mu'}$$

# Contenido

- 1 Introducción
- 2 Problema
- 3 Objetivo
- 4 Conocimientos previos
- 5 Herramientas Utilizadas
- 6 Análisis del Entorno
- 7 Metodología y desarrollo
- 8 Entrenamiento
- 9 Resultados
- 10 Conclusiones
- 11 Trabajos futuros

# Entrenamiento



# Contenido

1 Introducción

2 Problema

3 Objetivo

4 Conocimientos previos

5 Herramientas Utilizadas

6 Análisis del Entorno

7 Metodología y desarrollo

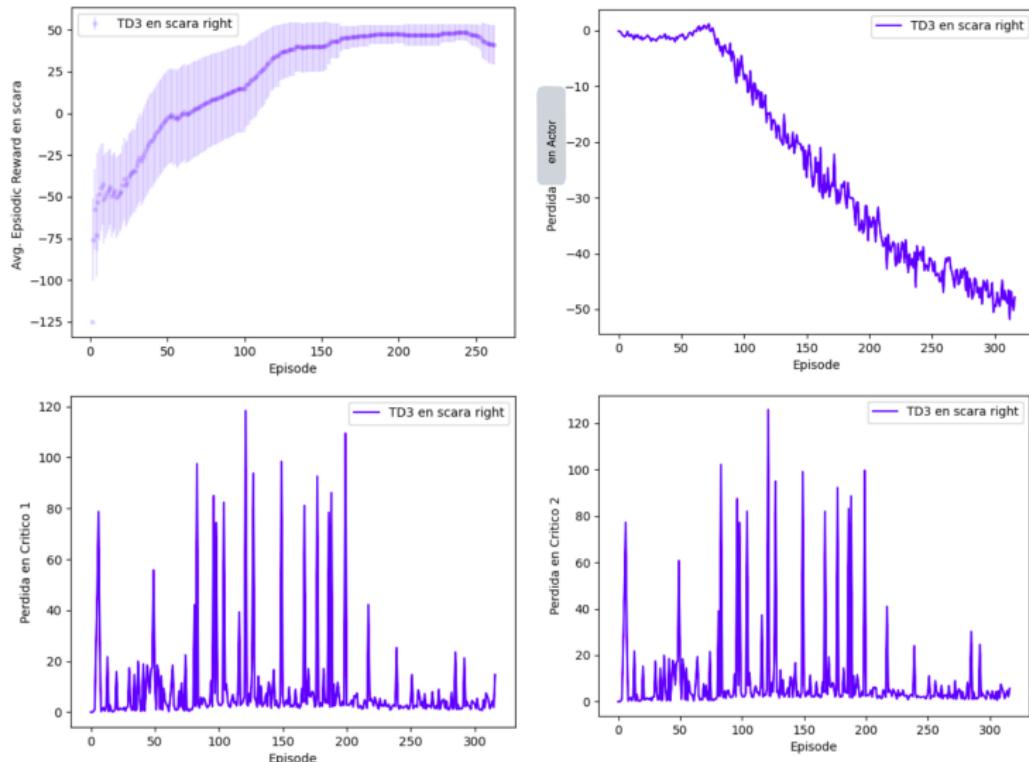
8 Entrenamiento

9 Resultados

10 Conclusiones

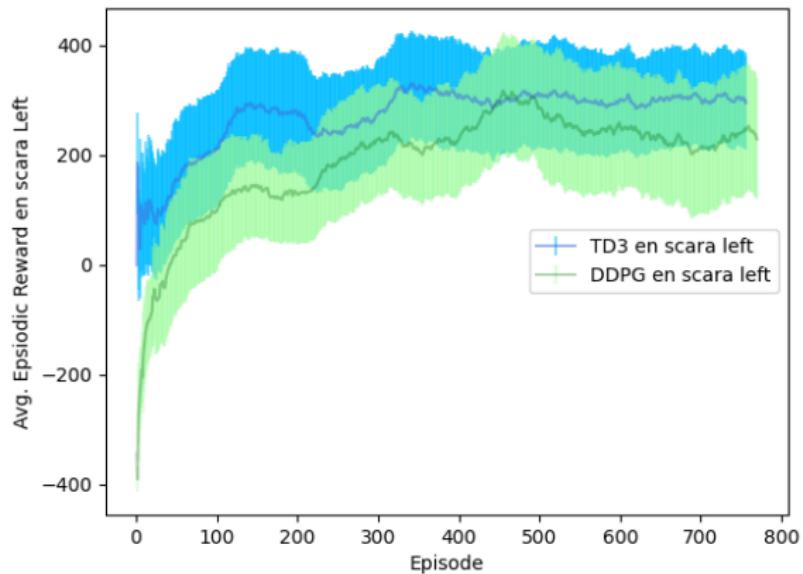
11 Trabajos futuros

# Resultados Defensa en Scara Right



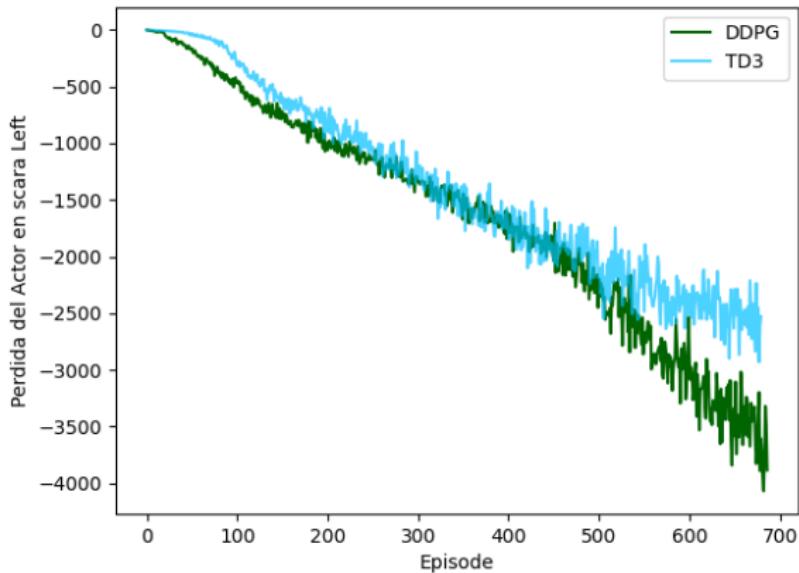
Figura

# Resultados Defensa y Ataque en Scara Left



Figura

# Resultados Defensa y Ataque en Scara Left



Figura

# Contenido

1 Introducción

2 Problema

3 Objetivo

4 Conocimientos previos

5 Herramientas Utilizadas

6 Análisis del Entorno

7 Metodología y desarrollo

8 Entrenamiento

9 Resultados

10 Conclusiones

11 Trabajos futuros

# Conclusiones

- Se logró construir un entorno Pybullet desde cero compatible con OpenAI gym y que cumple con las exigencias que este proyecto amerita.
- Se logró comprender como funcionan los métodos DDPG y TD3 por dentro, y la importancia de las políticas deterministas para poder admitir trabajar con acciones continuas.
- TD3 es superior a DDPG, teniendo una puntuación mayor al alcanzar la convergencia y obteniendo mas victorias al enfrentarse en juego. Resultados indiscutibles que lo hacen el modelo ganador, aunque era lo esperado se pudo corroborar tal desempeño.

# Contenido

1 Introducción

2 Problema

3 Objetivo

4 Conocimientos previos

5 Herramientas Utilizadas

6 Análisis del Entorno

7 Metodología y desarrollo

8 Entrenamiento

9 Resultados

10 Conclusiones

11 Trabajos futuros

# Trabajos futuros

- Mejora de Hiperparámetros
- Añadiendo mayor dimensionalidad a los Estados