



Aprendizaje por Refuerzo Profundo en un Robot Cuadrúpedo

Joaquín Mariano Piñeiro, Alejo Domínguez Nimo

Proyecto Final R6051 - Universidad Tecnológica Nacional - FRBA

jpineiro@frba.utn.edu.ar adomnguez@frba.utn.edu.ar

1. Introducción

Los robots cuadrúpedos como Spot de Boston Dynamics o Go2 de Unitree logran movimientos avanzados gracias a sofisticados y costosos actuadores y sistemas de control, lo que limita su accesibilidad y flexibilidad para diversas aplicaciones. Esto plantea la necesidad de una alternativa más asequible basada en hardware convencional, capaz de ejecutar movimientos eficientes sin sistemas de control altamente complejos.

Este trabajo desarrolla un robot cuadrúpedo de 12 grados de libertad que emplea IA, específicamente el algoritmo Multiple Objective Soft Actor Critic, permitiéndole aprender a desplazarse sin una caminata programada. Su diseño económico utiliza partes impresas en 3D y servomotores, a diferencia de las opciones actuales con materiales y motores más costosos. Además, la incorporación de sensores como LIDAR ampliaría sus capacidades de exploración y adaptación a distintas tareas.

2. Metodología

Primero se realizó el algoritmo de aprendizaje por refuerzo descrito en el paper presentado por los integrantes del proyecto en el **“2024 IEEE Biennial Congress of Argentina (ARGENCON)”**. Posteriormente se diseñó la estructura 3D del robot en SolidWorks minimizando el momento de inercia de las extremidades.



Paralelamente se implementó en una NUCLEO-F412ZG un sistema de control que manipula los servomotores de cada articulación con un control PI, una máquina de estados para supervisar la posición angular y un filtro de mediana para acondicionar las señales eléctricas proveniente de los motores, de las cuales se obtiene la posición angular. Además, se implementó una IMU con un acelerómetro y un giróscopo fusionados con un filtro de Kalman en una XIAO-nRF52840.

Una vez obtenido el modelo 3D, se lo exportó en un archivo URDF (Universal Robot Description Format) para poder entrenar la IA en un entorno de simulación llamado Webots. En la simulación se tuvieron en cuenta varios aspectos para acercarla lo más posible a la realidad, como:

- Modelo real de los servomotores (planta del motor DC y control PID).
- Delay del Filtro de Mediana.
- Idéntico control PI, máquinas de estados y filtro de Kalman que en el firmware de la NU-

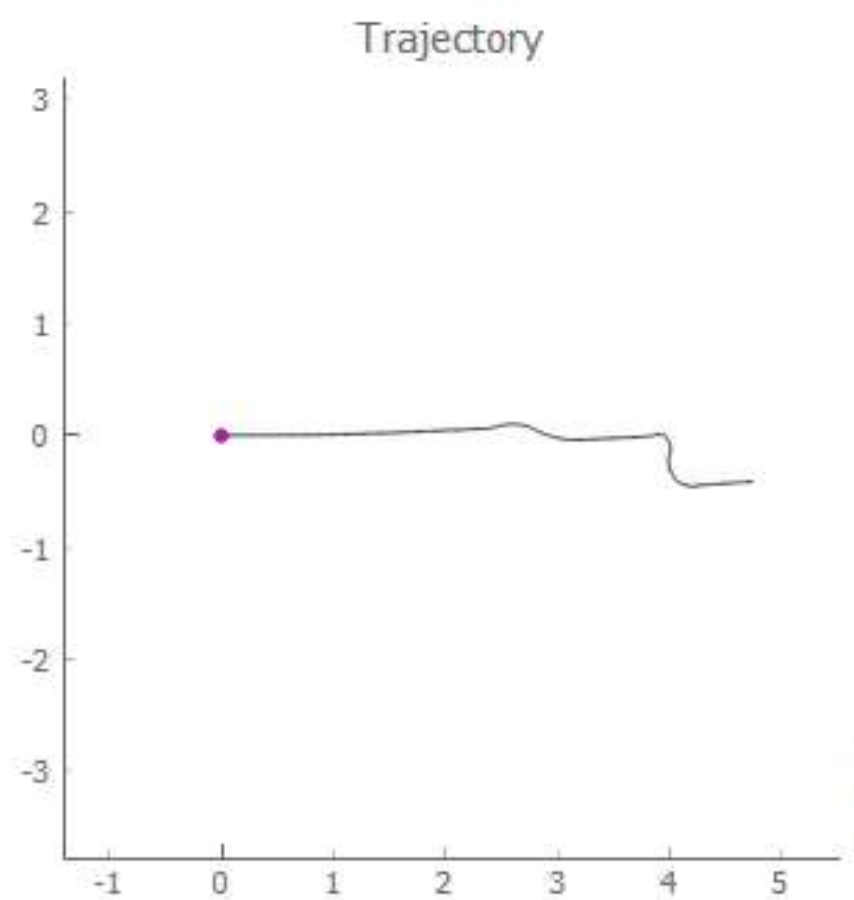
CLEO y XIAO respectivamente.

- Masas, centros de masa y matrices de inercia de cada cuerpo rígido del robot real.

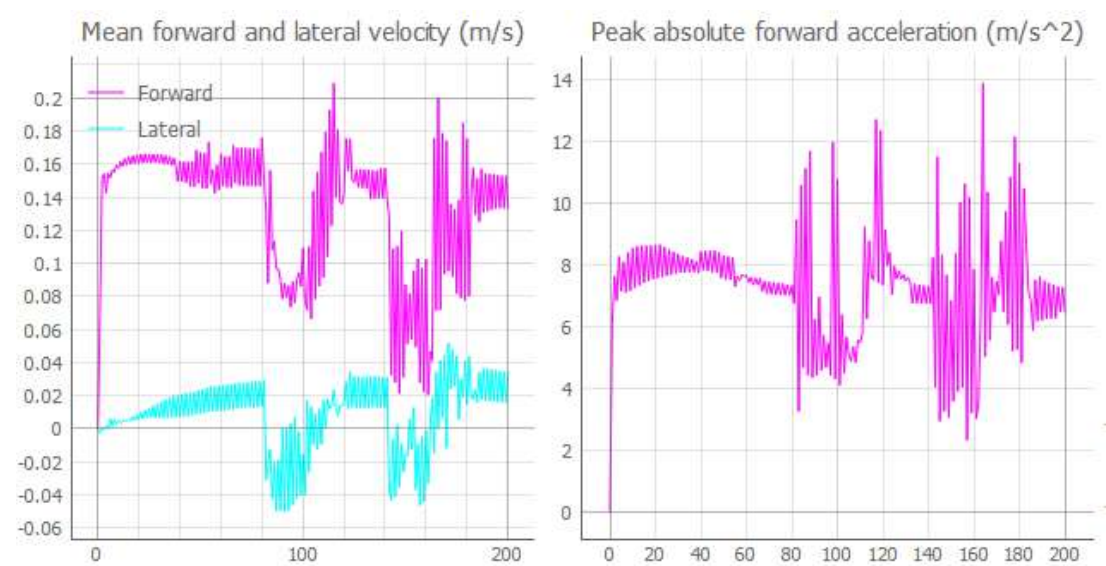
Finalmente se realizaron sucesivas iteraciones para evaluar y mejorar la caminata obtenida del entrenamiento en el simulador, en el robot real.

3. Resultados

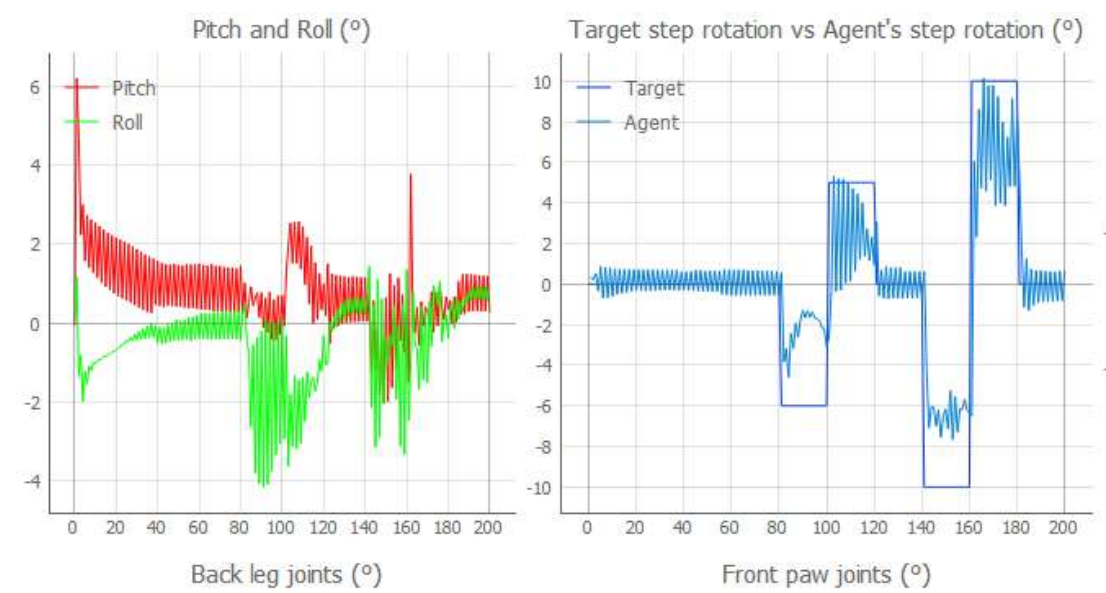
En los QR se muestran videos de distintas caminatas. En el primero de ellos se puede visualizar la caminata obtenida luego del entrenamiento en el entorno simulado. Durante esta prueba se le indica al agente que se mantenga derecho durante los primeros 80 pasos, y luego cada 20 pasos rote con distintos ángulos objetivo: 6° hacia la derecha, 5° hacia la izquierda, 0° nuevamente, 10° hacia la derecha y finalmente 10° hacia la izquierda. La siguiente figura muestra la trayectoria en metros realizada durante el test en la simulación.



El agente logra mantener una velocidad constante de $0.16 \frac{m}{s}$ durante el trayecto recto y luego reduce su velocidad en los momentos de rotación:



Además, mantiene la espalda considerablemente recta ya que la inclina solamente entre de 2° y 4° en pitch durante el trayecto recto y en roll en el momento de rotación, respectivamente. Por otro lado es capaz de seguir la rotación target en todo momento y cambiar rápidamente de dirección al pasar de -10° a 10°:



La caminata en el robot real se puede observar, tanto en el aire como apoyado sobre el suelo, en los QRs.



Caminata en simulación



Caminata real en aire



Caminata real en suelo



Caminata programada

4. Conclusión

En la literatura referida al aprendizaje por refuerzo, existe una problemática conocida y frecuentemente abordada llamada “Reality Gap”. La misma se refiere a la degradación del desempeño de la inteligencia artificial entrenada en una simulación al transferirla al robot físico, debido a las diferencias entre el entorno simulado y el real. En el contexto actual, se estudian diversas herramientas para intentar mitigarla. Contemplando esta problemática en el estado del arte, el resultado obtenido es satisfactorio teniendo en cuenta el tiempo y recursos a disposición.

Por otro lado, al diseñar la caminata programada fue más evidente que si bien el robot es capaz de caminar, es importante tener en cuenta la ubicación del centro de masa en todo momento para lograr estabilidad.

Este trabajo sienta las bases para el estudio de métodos para mitigar el “Reality Gap”:

- Estudiar la forma de hacer factible que la última parte del entrenamiento sea realizado en la realidad, para ajustar la caminata base obtenida en la simulación.
- Funciones de política más transferibles a la realidad: aleatorización de parámetros físicos, con posible información al agente de dichos parámetros a fin de evitar degradación en el comportamiento.
- Rediseño de la estructura con la posibilidad de aumentar el momento de inercia del chasis y disminuir la inclinación del cuerpo.
- Evaluar otras estrategias para una simulación más realista del robot.
- Diseño de funciones de recompensa que generen caminatas más fáciles de transferir a la realidad, minimizando el tiempo en el que las patas están en el aire.