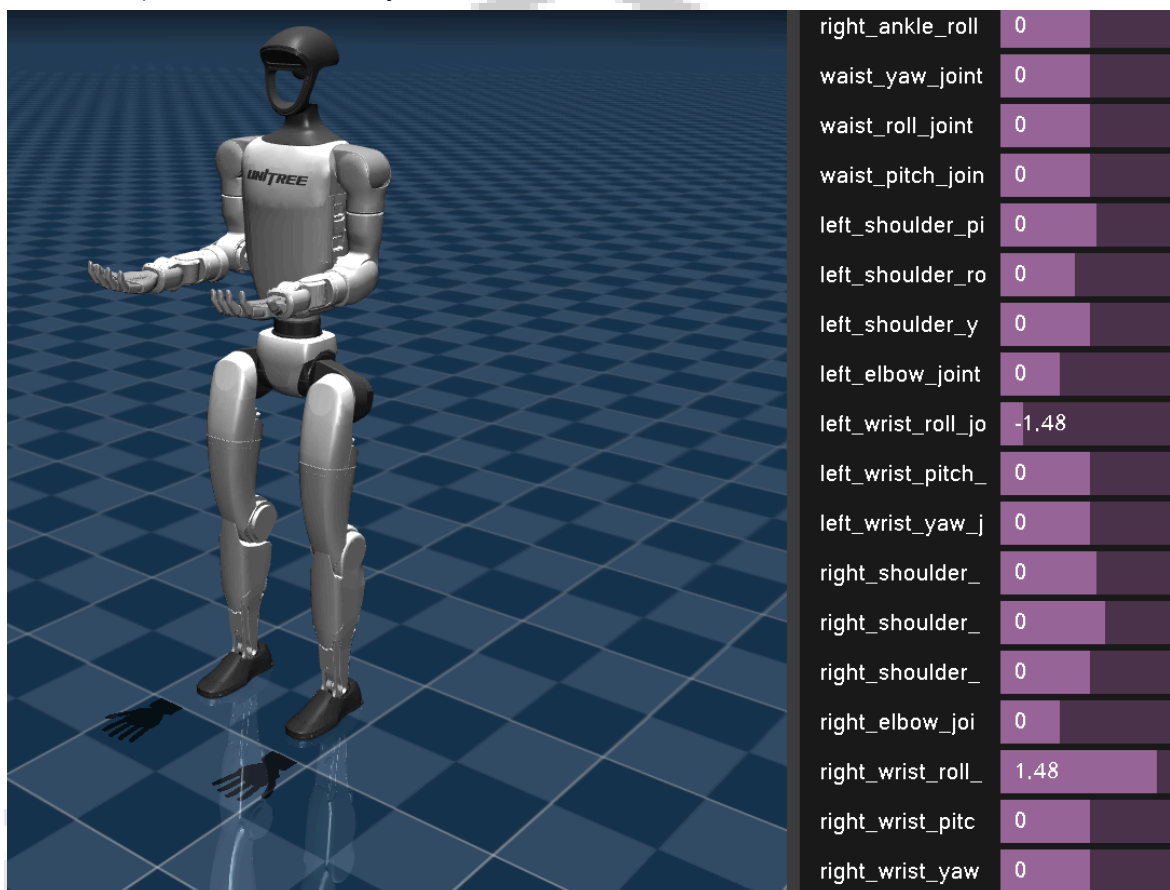


## 1. Pruebas del 10 de abril

Posición que se va a manejar



Los archivos txt están estructurados de la siguiente manera:

Para articulaciones de los brazos tienen nombre de la articulación y posición:

```
ejemploTrayectoria.txt
1 LeftShoulderPitch 0.0
2 LeftShoulderRoll 0.0
3 LeftShoulderYaw 0.0
4 LeftElbow 0.0
5 LeftWristRoll -1.48
6 LeftWristPitch 0.0
7 LeftWristYaw 0.0
8 RightShoulderPitch 0.0
9 RightShoulderRoll 0.0
10 RightShoulderYaw 0.0
11 RightElbow 0.0
12 RightWristRoll 1.48
13 RightWristPitch 0.0
14 RightWristYaw 0.0
15 WaistYaw 0.0
16 WaistRoll 0.0
17 WaistPitch 0.0
18
```

Para trayectoria predefinida donde esta la dirección y la duración del movimiento:

```
*ejemploTrayectoria.txt
1 adelante 2.5
2 rotar_izq 1.0
3 izquierda 3
4 atras 2
5 derecha 1.5
6
```

1. Prueba movimientos articulares superiores con caja para análisis de torques con diferentes pesos de caja - estático y movimiento de caminata con el control remoto

Scripts:

- g1\_armsdk\_moveV4.py (Control de articulaciones superiores)
- g1\_arm\_sdk\_visualizer\_pos\_torque.py (Visualización en tiempo real de datos)

## Objetivo General

Esta herramienta dual permite:

- Controlar de forma precisa y flexible las articulaciones superiores del robot cuadrúpedo G1 de Unitree utilizando la SDK2.
- Visualizar en tiempo real los valores de posición y torque estimado de cada articulación para monitoreo, análisis y depuración.

Ideal para sesiones de prueba, validación de movimientos programados y exploración de capacidades articulares del robot.

## Script 1: g1\_armsdk\_moveV4.py

### Objetivo del Script

El script permite controlar de forma interactiva las articulaciones superiores del robot G1 a través del canal arm\_sdk de la SDK2.

Soporta dos modos de operación:

- Modo interactivo manual: el usuario ingresa posiciones articulares deseadas (en radianes) desde la consola.
- Modo lectura desde archivo: permite cargar secuencias de movimiento desde un archivo .txt.

Se incluyen plantillas para facilitar este modo.

### Requisitos Previos

Antes de ejecutar el script, asegúrate de cumplir con los siguientes puntos:

- Conexión Ethernet activa entre tu PC y el robot G1.

(IP fija configurada correctamente, e.g. 192.168.123.X).

- Robot encendido y en modo "Main Operation Control".
- Arnés de seguridad colocado al robot.
- SDK2 correctamente instalada y funcional.
- Python 3 y dependencias instaladas (numpy, csv, etc.).

Instalar:

### **pip install pyqtgraph PyQt5**

- Se recomienda tener la versión G1 29DoF.
- El script genera un archivo .csv automáticamente — verifica permisos de escritura.

### **Cómo Ejecutar el Script**

Desde terminal:

**python3 g1\_arm\_sdk\_moveV4.py <nombre\_de\_interfaz>**

### **Funcionamiento General**

#### **1. Inicialización:**

- Conexión con los canales DDS (lowstate, arm\_sdk).
- Se verifica disponibilidad de LocoClient.
- Se inicia suscripción al estado del robot.

#### **2. Lectura del Estado:**

- El script detecta las posiciones articulares actuales como punto de partida.
- Cada (500 ciclos) se guarda el estado en un .csv con:

timestamp,  $q[i]$ ,  $\tau[i]$ .

#### **3. Interfaz Interactiva:**

- El usuario ingresa el valor deseado de cada articulación.
- Enter sin valor  $\Rightarrow$  asume 0.0 rad.
- Comando exit  $\Rightarrow$  termina el programa.

#### **4. Movimiento Interpolado:**

- Se usa interpolación sinusoidal entre la posición actual y deseada.
- Duración por defecto: 5 segundos (configurable).
- Se valida que se alcanzó el objetivo articular (dentro de tolerancia).

#### **5. Liberación del Control:**

- Se libera el canal arm\_sdk suavemente.
- Se cierra y guarda el archivo .csv.

## Salida de Datos

Se genera automáticamente un archivo .csv con nombre:

**data\_g1\_YYYYMMDD\_HHMMSS.csv**

Columnas:

**timestamp**

**$q_0, \tau_0, q_1, \tau_1, \dots, q_N, \tau_N$**

Para todas las articulaciones superiores disponibles.

**Script 2: g1\_arm\_sdk\_visualizer\_pos\_torque.py**

### Objetivo del Script

Este script permite visualizar en tiempo real los datos de posición y torque de las articulaciones superiores del G1 que se están registrando en el .csv generado por g1\_armsdk\_moveV4.py.

Permite seleccionar qué articulaciones visualizar y facilita el análisis post-ejecución o en simultáneo.

### Requisitos Previos

Python 3

- Instaladas las librerías: PyQt5, pyqtgraph
- Un archivo .csv generado previamente o en tiempo real por g1\_armsdk\_moveV4.py.

### Cómo Ejecutar

Desde terminal:

**python3 g1\_arm\_sdk\_visualizer\_pos\_torque.py**

Luego se solicita en consola:

Ingrese la ruta del archivo .csv:

**Ejemplo: data\_g1\_20250409\_183200.csv**

### Funcionamiento General

- El script abre una ventana gráfica con dos paneles:

- Posición ( $q$ ): curva continua.
- Torque estimado ( $\tau$ ): línea punteada.
- El usuario puede seleccionar qué articulaciones visualizar usando checkboxes.
- Los gráficos se actualizan automáticamente cada 50ms ( $\sim 20\text{Hz}$ ).
- Zoom sincronizado entre los dos gráficos.

### Uso en Simultáneo

Para ejecutar ambos scripts en paralelo:

Abre una terminal para el control:

```
python3 g1_arm_sdk_moveV5.py eth0
```

Abre otra terminal y lanza el visualizador:

```
python3 g1_arm_sdk_visualizer_pos_torque.py
```

En el visualizador, introduce el nombre del .csv que se está generando por el primer script (puede estar en la misma carpeta).

Ahora es posible poner diferentes pesos sobre los brazos del robot y evaluar el comportamiento del torque de las articulaciones.

Requiere permisos de lectura concurrente sobre el archivo.

El protocolo de evaluación seguirá el poner distintos pesos con el robot estático y luego en movimiento vía control para evaluar el soporte según el torque y ejecución de la marcha.

## 2. Prueba movimientos articulares superiores con caja para análisis de torques con caja y peso - movimiento de caminata trayectoria predefinida

### Objetivo del Script

El script `g1_arm_sdk_moveV5.py` permite controlar de forma interactiva las articulaciones superiores del robot cuadrúpedo G1 de Unitree, utilizando la SDK2 y el canal `arm_sdk`. Soporta dos modos de operación para ingresar los movimientos:

- Modo interactivo manual: el usuario escribe las posiciones articulares deseadas directamente en consola.
- Modo lectura desde archivo: se leen secuencias articulares desde un archivo .txt, permitiendo movimientos preprogramados o pruebas repetibles. (Se incluye unas plantillas de archivos txt)

## Requisitos Previos

Antes de ejecutar el script, asegúrate de cumplir los siguientes puntos:

- Conexión Ethernet activa entre tu PC y el robot G1 (Configuración de IP listai).
- Robot encendido y en modo "Main Operation Control".
- Arnés de seguridad colocado al robot para prevenir caídas durante movimientos inesperados.
- SDK2 correctamente instalada y funcional en tu entorno Python.
- Se recomienda tener la versión G1 29DoF, ya que algunas articulaciones no están presentes en la versión 23DoF.

El script genera un archivo CSV automáticamente para guardar los datos articulares. Verifica que tengas permisos de escritura en el directorio.

## Cómo Ejecutar el Script

Desde una terminal con el entorno adecuado:

```
python3 g1_armsdk_moveV5.py NOMBRE_INTERFAZ
```

## Funcionamiento General del Script

### → Inicialización:

Se establece conexión con los canales DDS de la SDK2 (lowstate y arm\_sdk).

Se inicializa el LocoClient para confirmar disponibilidad de control de alto nivel.

Se inicia la suscripción al estado del robot para obtener lecturas en tiempo real.

### → Lectura del Estado:

Se recibe el estado completo del robot.

Se detectan automáticamente las posiciones articulares actuales como punto de partida.

Cada 500 ciclos, se registra en un archivo .csv la posición y torque estimado de las articulaciones superiores.

→ Interfaz Interactiva:

Se solicita al usuario ingresar valores deseados para cada articulación (en radianes).

Si se presiona Enter, se asume 0.0 rad para esa articulación.

El usuario puede salir escribiendo exit.

→ Movimiento Interpolado:

El movimiento entre la posición actual y la deseada se realiza con interpolación sinusoidal (más suave que lineal).

La duración del movimiento es configurable (por defecto, 5 segundos).

Se verifica que el robot haya llegado a la posición deseada dentro de una tolerancia.

En caso de no alcanzar la posición dentro del tiempo límite, se notifica.

→ Liberación del Control:

Al finalizar las pruebas o si el usuario detiene el script, se libera progresivamente el control.

Se desactiva el canal arm\_sdk suavemente para evitar cambios bruscos.

Se cierra el archivo .csv y se informa al usuario.

### Consideraciones de Seguridad

No dejar el robot sólo durante la ejecución del script. Se recomienda un observador humano.

### Salida de Datos

El script guarda automáticamente un archivo con nombre:

**data\_g1\_YYYYMMDD\_HHMMSS.csv**

Contiene: timestamp, posición q y torque tau para cada articulación superior del G1.