




Protocolo Técnico de Prueba para Control Combinado de Brazos y Locomoción – Robot G1 (Unitree)

Objetivo General: Ejecutar una secuencia controlada donde el robot G1 se acerque a una caja, realice el agarre mediante movimientos de brazo, transporte el objeto caminando, y lo deposite en un objetivo determinado. Se busca validar la integración brazo-locomoción mediante una ejecución segmentada, parametrizada y evaluable.

1. Preparación del Entorno

1.1 Disposición física del espacio

- Marcar claramente sobre el suelo:
 -  **Punto de inicio del robot**
 -  **Posición inicial de la caja**
 -  **Ubicación del objetivo final**
- Asegurar:
 - Caja con peso conocido (entre 0.5 kg y 1.5 kg)
 - Colocación del arnés de seguridad
 - Activación del modo **Main Operation Control**
 - Conexión de red activa y estable entre el G1 y la estación de control

1.2 Parametrización experimental de trayectorias

Objetivo: Determinar, mediante prueba controlada previa, los tiempos óptimos que el robot debe mantener para cada segmento de locomoción, según las distancias físicas reales entre puntos de interés.

Procedimiento de calibración:

1. Sin carga, ejecutar manualmente secuencias de caminata utilizando el script.
2. Medir distancias físicas entre:
 - Punto de inicio y posición de la caja
 - Posición de la caja y el objetivo final
3. Determinar tiempo requerido por segmento mediante observación y ajuste:
 - Ajustar tiempo por dirección (**adelante**, **rotar_izq**, etc.) hasta que el robot alcance físicamente el punto deseado.
4. Registrar los valores de tiempo óptimos para cada segmento en archivos **.txt**:
 - **trayectoria_fase1.txt** (acercamiento a la caja)
 - **trayectoria_fase3.txt** (camino al objetivo)

Ejemplo en archivo txt:

```
adelante 2.3  
rotar_der 0.8  
adelante 1.7
```

Importante: Esta calibración debe realizarse en la fase de preparación y validación inicial del entorno, antes de ejecutar la prueba completa.

2. Definición de Posturas con MuJoCo

Cargar el modelo del G1 en MuJoCo:

Si no se encuentra instalado: pip install mujoco

```
python -m mujoco.viewer
```

Arrastrar el robot al visualizador desde la carpeta, seleccionar pause y luego reset. Ahí se encontrará habilitado para mover las articulaciones.

Definir y almacenar posturas articulares para cada fase:

Fase	Descripción	Archivo recomendado
Acercarse	Postura lista para agarrar	fase1_approach.txt
Agarrar	Cierre de manos, contacto con la caja	fase2_grasp.txt
Transporte	Brazos replegados con objeto sostenido	fase3_hold.txt
Soltar	Postura para depositar la caja	fase4_release.txt

Formato del archivo .txt:

```

RightShoulderPitch 0.0
RightElbow          1.2
RightWristRoll      1.48
...
  
```

Se recomienda copiar manualmente los valores desde la interfaz visual de MuJoCo tras cada simulación.

3. Ejecución del Script **g1_armSDK_moveV5.py**

El script está diseñado para ejecutar las fases de movimiento en secuencia con intervención del usuario. Cada módulo es interactivo y modular.

Ejecución del script: **g1_armSDK_moveV5.py nombreInterfaz**

3.1 Movimiento a Posición Cero

Moviendo a posición cero... Presione Enter para continuar.

- Posición neutral de referencia para todos los brazos.
- Se pregunta al usuario si desea continuar con la caminata.

3.2 Ingreso de Postura Objetivo

¿Deseas ingresar posiciones manualmente o cargar desde archivo?

1. Ingresar manualmente

2. Cargar desde archivo .txt

Opción 1 – Ingreso manual

- Ingreso de cada articulación por separado.
- Puede cancelarse con **exit**.

Opción 2 – Desde archivo

- Se valida que los nombres de articulaciones estén completos y correctos.
- Utiliza archivos generados en la fase 2.

3.3 Movimiento del Brazo a Postura de Agarre

→ Iniciando movimiento...

- El robot ejecuta la postura cargada.
- Se solicita confirmación para proceder a la caminata.

3.4 Ejecución de Caminata (opcional)

¿Deseas que el robot ejecute una trayectoria ahora? (s/n)

Opción 1 – Ingreso manual:

- Dirección (**adelante**, **izquierda**, **rotar_der**, etc.)
- Tiempo en segundos

Opción 2 – Cargar desde archivo .txt:

Ejemplo `trayectoria_fase3.txt`:

adelante 2.5

rotar_der 1.0

adelante 1.5

La función `execute_trajectory_sequence()` recorrerá cada línea, mostrando un resumen antes de ejecutar.

3.5 Repetición / Retorno / Finalización

Después de cada movimiento o caminata:

1. Ingresar nueva posición objetivo

2. Volver a posición cero

Después de retornar a cero:

1. Ingresar otra posición objetivo

2. Liberar control y salir

El script finaliza limpiamente y libera el control del robot de forma segura.

4. Secuencia Recomendada de Ejecución

1. **fase1_approach.txt** → Movimiento de brazos (preparación para agarre)
2. **trayectoria_fase1.txt** → Caminata hacia la caja

3. **fase2_grasp.txt** → Postura de agarre (cerrar brazos) *evaluar si es necesaria fase 3
4. **fase3_hold.txt** → Repliegue de brazos para caminata
5. **trayectoria_fase3.txt** → Caminata al objetivo
6. **fase4_release.txt** → Postura para soltar la caja
7. **Posición cero** → Retorno al estado neutral
8. **Fin del experimento**

