UNIVERSIDAD DE DEUSTO

FACULTAD DE INGENIERÍA

HITO 2

ROBÓTICA

Docente(s):

Ignacio Fidalgo

Xabier Angulo

Joshue Manuel

Estudiante(s):

Iñigo Picón

Eduardo Larrinaga

Mikel Lambarri

Gorka Esteban

El nodo signal_capture_node, lo utilizamos para capturar y registrar datos relacionados con los estados articulares del brazo robótico UR3e.

1. Funcionalidad del Nodo

1. **Captura de Señales Articulares**: Este nodo se suscribe al topic /joint_states, donde se publican los datos de velocidad, posición y esfuerzo de las articulaciones del sistema robótico. Cada vez que recibe un mensaje, procesa la información y la almacena temporalmente en una lista mientras la captura sigue.
2. **Control de Captura**: Mediante el topic /capture_control, el nodo permite iniciar y detener la captura de datos. Recibe comandos numéricos (1 para iniciar y 0 para detener), lo que nos da un control sencillo para manejar la recopilación de información. Cada vez que capturamos datos asociamos la captura con un código único de trazabilidad.
3. **Registro en InfluxDB**: Una vez finalizada la captura, los datos recopilados se procesan y se registran en una base de datos InfluxDB alojada en AWS.
4. **Gestión de Datos Previos**: Antes de iniciar una nueva captura de datos, el nodo elimina todos los datos existentes en el bucket correspondiente de InfluxDB, asegurando que la base de datos esté limpia y evitando conflictos de datos.

2. Nodos con los que interactúa y sus topics

Nodo 1: capture_image

- **Función:** Procesar imágenes recibidas de una cámara y detectar la cantidad de dedos levantados.
- **Topics:**
    - **Suscripción:**
        - **/usb_cam/image_raw:** Recibe imágenes en tiempo real desde la cámara.
            - **Tipo de mensaje:** sensor_msgs/Image.
    - **Publicación:**
        - **/send_int:** Publica un número entero que representa el número detectado de dedos levantados.
            - **Tipo de mensaje:** std_msgs/Int32.

Nodo 2: control_robot

- **Función:** Recibe el número procesado por el nodo capture_image y lo utiliza para realizar un movimiento correspondiente en el brazo UR3e. Además, publica señales para el registro en la base de datos.
- **Topics:**
    - **Suscripción:**
        - **/send_int**: Recibe el número detectado por el nodo capture_image.
            - **Tipo de mensaje:** std_msgs/Int32.
    - **Publicación:**
        - **/capture_control:** Publica una señal (1 antes del movimiento y 0 después) para marcar el inicio y fin de una trayectoria.

- **Tipo de mensaje:** std_msgs/Int32.

Nodo 3: signal_capture_node

- **Función:** Registra datos de las trayectorias realizadas por el brazo robótico en una base de datos InfluxDB.
- **Topics:**
  - **Suscripción:**
    - **/capture_control:** Recibe señales de inicio y fin de trayectoria.
      - **Tipo de mensaje:** std_msgs/Int32.
    - **/joint_states:** Registra los estados de las articulaciones del robot (posiciones, velocidades y esfuerzos).
      - **Tipo de mensaje:** sensor_msgs/JointState.

3. Integración con el Proyecto

El nodo signal_capture_node se conecta directamente con otros nodos del sistema a través de los topics mencionados.

Comunicación entre control_robot **e** signal_capture_node

- **Canal:** /capture_control (Topic).
- **Tipo de Comunicación:** Síncrono.
- **Detalle:** El nodo control_robot publica señales antes y después de realizar una trayectoria en el topic /capture_control. Estas señales son utilizadas por el nodo influx para marcar los límites de las trayectorias registradas. Manda un 1 para que empiece a registrar información y un 0 para dejar de registrar.

Comunicación entre el Robot e signal_capture_node

- **Canal:** /joint_states (Topic).
- **Tipo de Comunicación:** Asíncrona.
- **Detalle:** Los datos del estado de las articulaciones del robot (posiciones, velocidades y esfuerzos) se publican continuamente en el topic /joint_states. El nodo influx registra estos datos para análisis posteriores.

4. Trayectorias

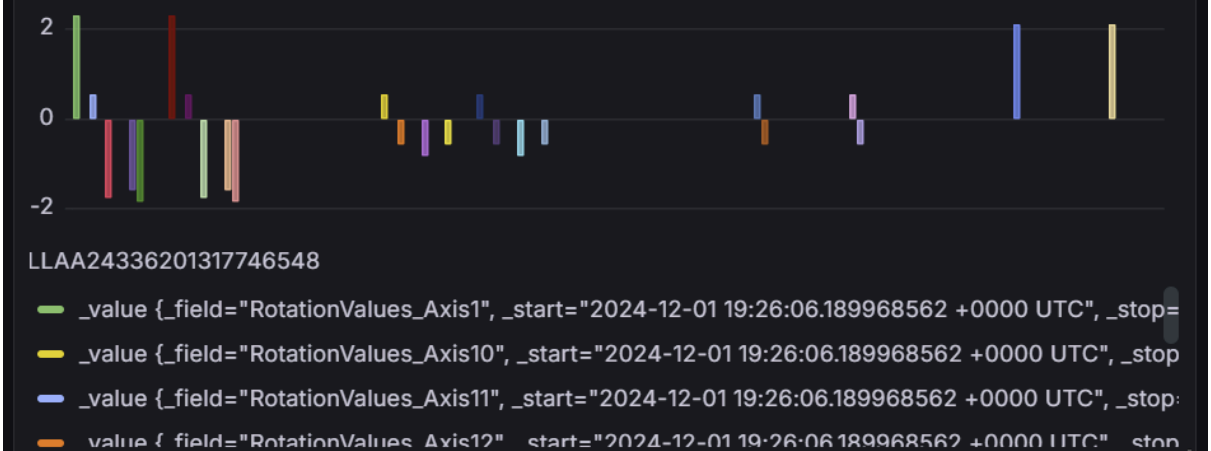
Hemos añadido 6 trayectorias. Tres de ellas son trayectorias que dibujan números que fueron creados para el proyecto. Las otras tres las hemos creado manualmente para que fallen. La primera falla porque colisiona con el obstáculo creado para representar el suelo y las otras dos fallan porque le hemos indicado que vaya a una distancia más lejana de lo que puede el brazo robótico.

5. Visualización de los datos

Los datos registrados en InfluxDB son posteriormente accedidos y analizados mediante Grafana. Aquí están los resultados obtenidos:

Todos

Máximos y Mínimos de Rotación por Eje



Conteo de Datos

224	224	224	224	224	224	224	224	224
224	224	224	224	224	224	224	224	224
224	224	224	224	224	224	224	224	224
224	224	224	224	224	224	224	224	224

6. Alarmas (opcional)

Esta alerta evalúa los datos en el bucket "Grupo_1" y verifica si algún valor de velocidad en cualquiera de los ejes (VelocityValues_Axis1, VelocityValues_Axis2, etc.) supera el umbral de 10 rad/s en los últimos 5 minutos.