

Robótica

Robótica de Desarrollo, Intro a ROS No.3

Integrantes:

Daniel Felipe Segura Rincon - dsegurar@unal.edu.co

Juan Andres Barrera Rodriguez - jbarreraro@unal.edu.co

Universidad Nacional de Colombia

2023-2

Metodología:

1. **Introducción:** El objetivo de este laboratorio es conocer y familiarizarse con los conceptos básicos de ROS (Robotic Operative System), usar sus comandos fundamentales, conectar nodos de ROS con Matlab y realizar conexión inicial de motores Dynamixel con ROS.
2. **Materiales y equipos:**
 - a. Ubuntu preferible 20.04 LTS.
 - b. MATLAB 2015b o superior.
 - c. Robotics toolbox de matworks.
 - d. Computador :v jajajaja
3. **Procedimiento:**
 - a. Familiarizarse con los comandos de mayor uso para la consola de Linux.
 - b. **Conexión de ROS con Matlab:** Se siguen todos los pasos indicados en el documento encontrado en la plataforma moodle y se realiza el código respectivo en Matlab para modificar la posición de la tortuga en el programa.
 - c. **Utilizando Python:** Se siguen todos los pasos indicados en el documento encontrado en la plataforma moodle y se realiza el código respectivo en python para modificar la posición y orientación de la tortuga, utilizando el teclado, con las respectivas especificaciones:
 - i. Se debe mover hacia adelante y hacia atrás con las teclas W y S.
 - ii. Debe girar en sentido horario y antihorario con las teclas D y A.
 - iii. Debe retornar a su posición y orientación centrales con la tecla R.
 - iv. Debe dar un giro de 180° con la tecla ESPACIO .

Resultados:

1. **Conexión de ROS con Matlab:** El código utilizado para desarrollar el ejercicio se encuentra en el repositorio de GitHub con el nombre "Laboratorio3.m ". Para este ejercicio fue posible leer la posición actual de la tortuga exitosamente. Sin embargo, el envío del mensaje con una nueva posición no fue posible.

| scan | posMsg |
|------------------|--------------|
| 1x1 Pose | |
| Property | Value |
| MessageType | 'turtlesim/F |
| X | 5.5444 |
| Y | 5.5444 |
| Theta | 0 |
| LinearVelocity | 0 |
| AngularVeloci... | 0 |

Figura 1. Lectura de la posición de la tortuga.

2. **Utilizando Python:** El código utilizado para desarrollar el ejercicio se encuentra en el repositorio de GitHub con el nombre "myTeleopKey.py". En este ejercicio, se logro controlar satisfactoriamente la tortuga usando las teclas 'a','s','d' y 'w'. Sin embargo, los controles de las teclas 'r' y 'espacio' no se consiguieron hacer.



Figura 2. Prueba movimiento de tortuga con python.

Análisis:

1. **Discusión:** Los resultados obtenidos en el laboratorio muestran que logramos con éxito conectar ROS con MATLAB y utilizar Python para controlar una tortuga simulada. Esta conexión resulta esencial en aplicaciones de robótica, donde la capacidad de integrar ROS con otras herramientas, como MATLAB, amplía significativamente las posibilidades de control y programación de robots. Nuestros objetivos de familiarizarnos con estos conceptos y herramientas se cumplieron de manera efectiva.

2. **Limitaciones:** Sin embargo, es importante reconocer algunas limitaciones en nuestro enfoque. En primer lugar, la necesidad de la Robotics Toolbox de MathWorks pueden restringir la accesibilidad de este enfoque a ciertos usuarios. Además, el éxito de la conexión ROS-MATLAB puede depender de la compatibilidad del sistema operativo Ubuntu y sus versiones, lo que podría generar complicaciones en configuraciones diferentes. También, la eficacia de la simulación en Python podría verse afectada por la velocidad de procesamiento y la capacidad del computador.

Conclusiones:

- Este laboratorio nos permitió adquirir conocimientos importantes en el uso de ROS, MATLAB y Python lo que proporciona una base sólida para futuras investigaciones y aplicaciones en el campo de la robótica.
- Se logró hacer la conexión ROS con MATLAB y controlar una tortuga simulada con Python, lo que es fundamental en el desarrollo y control de sistemas robóticos.
- Se debe tener en cuenta las limitaciones mencionadas, que podrían afectar la aplicabilidad en diferentes configuraciones.