

Práctica 5 – Robot Operating System

Instituto Tecnológico Autónomo de México
Departamento Académico de Sistemas Digitales
Laboratorio de Principios de Mecatrónica

Primavera 2022

1 Objetivos

- Reconoces las características de ROS como una plataforma de comunicación, control y simulación para aplicaciones de Robótica.
- Lograr un acercamiento a la simulación de agentes móviles en un entorno especializado.
- Realizar el modelado y control de un robot móvil de tipo diferencial.

2 Plataforma

- ROS Kinetic Kame <http://wiki.ros.org/kinetic/Installation> o superior.
- Ubuntu Xenial (16.04 LTS) o superior.

3 Configuración de paquetes

```
~/catkin_ws$ cd ~/catkin_ws
~/catkin_ws$ cd src
~/catkin_ws$ catkin_create_pkg paquete std_msgs rospy roscpp
~/catkin_ws$ cd ~/catkin_ws
~/catkin_ws$ catkin_make
~/catkin_ws$ . ~/catkin_ws/devel/setup.bash
~/catkin_ws$ source /opt/ros/kinetic/setup.bash
~/catkin_ws$ catkin_make
~/catkin_ws$ source devel/setup.bash
~/catkin_ws$ roscd paquete
~/catkin_ws$ mkdir scripts
~/catkin_ws$ cd scripts
~/catkin_ws$ ls
~/catkin_ws$ chmod +x node.py
~/catkin_ws$ cd ~/catkin_ws
~/catkin_ws$ catkin_make
~/catkin_ws$ . ~/catkin_ws/devel/setup.bash
```

4 Procedimientos

4.1 Ejecución ROS

Ejecutar ROSCORE a través de una terminal del sistema (Ctrl+Alt+T).

```
~/ $ roscore
```

4.2 Turtlesim

Ejecutar el nodo del simulador Turtlesim (del paquete turtlesim) en una segunda ventana/pestaña de terminal [Ctrl+Shift+T] (Figura 1).

```
~/ $ rosrunc turtlesim turtlesim_node
```

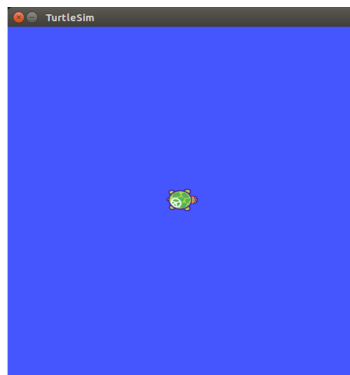


Figura 1: Simulador Turtlesim.

4.3 Roservices

En una tercera terminal, utilizar los servicios para reemplazar al vehículo en una nueva posición y orientación (Figura 2).

```
rosservice call /kill 'turtle1'  
rosservice call /spawn 1 1 1.57 'turtle1'
```

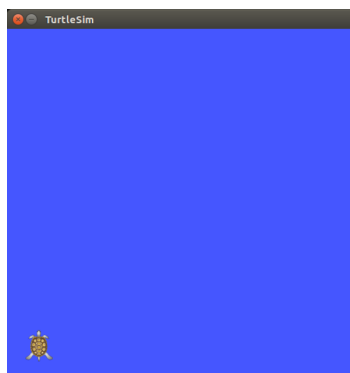


Figura 2: Estableciendo posición inicial.

4.4 Ejecución de un nodo

Trasladarse al espacio de trabajo y ejecutar un nodo para automatizar el desplazamiento del vehículo, en este caso una línea recta.

```
~/catkin_ws$ cd ~/catkin_ws
~/catkin_ws$ catkin_make
~/catkin_ws$ . ~/catkin_ws/devel/setup.bash
~/catkin_ws$ rosrn paquete node.py
```

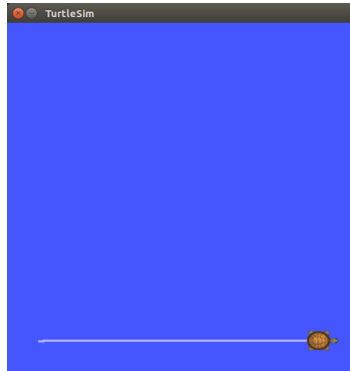


Figura 3: Ejecución de nodo.

4.5 Trayectoria

Implementar las modificaciones pertinentes al algoritmo Go-to-goal para completar una trayectoria rectangular.

4.6 Trayectoria

Implementar las modificaciones pertinentes al algoritmo Go-to-goal para completar una trayectoria circular.

Referencias Recomendadas

ROS <https://www.ros.org>

ROS wiki <https://www.wiki.ros.org>