

# Programa para movimientos de un robot

Diego Pozo 149392

September 1, 2016

## Abstract

Un reporte sobre el primer proyecto de la clase de robótica, en el cual se aprendió a utilizar ROS para crear un programa que simulara el movimiento de un robot.

## 1 Introducción

En este proyecto se aprendió a utilizar el middleware ROS para crear un programa interactivo con el usuario. El programa pide al usuario coordenadas y, junto con el programa "turtlesim", crea una simulación, en la cual una tortuga se mueve hacia el punto indicado, tomando el tiempo necesario y terminando en la pose requerida.

## 2 Marco teórico

### 2.1 ROS

ROS (Robot Operative System) es un middleware que se basa en Open Source. Es una herramienta muy útil para la programación de robótica, pues es altamente utilizado a lo largo del planeta. Por esto mismo, ROS contiene una gran variedad de paquetes y comandos que resultan de gran ayuda al momento de crear un programa. El uso de ROS va desde la industria hasta la academia, incluyendo a personas que programan como pasatiempo.

### 2.2 Trayectorias de robots

Las trayectorias son un tema muy importante en la robótica, pues el desarrollo de robots se centra cada vez más en robots móviles y en robots autónomos. Ésto termina en la necesidad de que los robots sean capaces de calcular y ejecutar trayectorias de diversas complejidades. Un concepto complementario para la trayectoria es el de la pose, que se refiere a la orientación del robot dentro de su espacio.

## 3 Experimentos

A lo largo de la creación del programa pasé por muchas ideas diferentes. Como primer intento, el plan fue encontrar el ángulo de diferencia entre la tortuga

y el punto deseado, girar hasta encontrarlo, avanzar en esa dirección con la velocidad deseada, y luego girar de nuevo hasta alcanzar la pose. Sin embargo, esto no es un método muy eficiente, pues requiere dividir la trayectoria en tres partes, haciendo el movimiento "no suave". Además, dicho método NO reducía la complejidad de los cálculos de manera significativa. Por estas razones se optó por seguir un código encontrado en un tutorial, el cual generaba una trayectoria suave que mezclaba la rotación y el avance de la tortuga de manera simultánea.

## 4 Conclusiones

Aunque al final se logró crear el programa con todos los requerimientos, hay muchas partes que se podrían mejorar. Primero que nada el punto del tiempo indicado por el usuario, pues no es muy exacto. Además de eso existen métodos que no son utilizados del todo en el código, pues fueron creados para casos un poco más generales, y por lo que podrían eliminarse para aumentar la eficiencia de un programa más específico como éste. La lección más valiosa que obtuve de este proyecto es la utilidad de la programación comunitaria, pues a lo largo del mundo muchas personas pueden encontrarse con obstáculos similares y puede ahorrar tiempo y problemas al cooperar, compartiendo diferentes códigos, problemas, errores y