



Tecnológico de Monterrey

Campus Puebla

Materia

Fundamentación de Robótica TE3001B

Tema

Challenge II

Integrantes

José Jezarel Sánchez Mijares A01735226

Frida Lizett Zavala Pérez A01275226

Antonio Silva Martínez A01173663

Fecha

Abril 25 2023

Contenidos

Challenge II	0
Resumen	2
Objetivos	2
Introducción	2
Solución del problema	3
Resultados	4
Conclusiones	4
Referencias	4

Resumen

El documento de investigación propone desarrollar las habilidades de los estudiantes en el manejo de ROS (Robot Operating System) y Gazebo, a través de una actividad que involucra la interacción con un robot llamado Puzzlebot. El objetivo de la actividad es aumentar el nivel de interacción y la implementación de funciones disponibles en ROS, así como diseñar un controlador P, PI o PID para minimizar el error de posición del robot.

Objetivos

Los principales objetivos de este reporte son los siguientes:

1. Comprender los requisitos técnicos necesarios para desarrollar el reto planteado, como la creación de una carpeta en GitHub, el uso de códigos para la creación de nodos, archivos launch y config, entre otros aspectos.
2. Justificar la importancia y el valor del desarrollo del reto en términos de la aplicación práctica de la tecnología de robótica en el mundo real.
3. Describir el proceso de desarrollo del reto, incluyendo la

identificación y resolución de problemas, las decisiones de diseño tomadas y los métodos utilizados para evaluar el rendimiento del robot.

4. Presentar las evidencias del funcionamiento del robot, incluyendo dos videos que muestren su desempeño en el simulador Gazebo y en el robot físico.
5. Proporcionar información sobre los equipos que participaron en la realización del reto, incluyendo los nombres de los integrantes de cada equipo.
6. Compartir el link del repositorio en forma individual y en la plataforma Canvas para su posterior evaluación.
7. Evaluar el éxito del proyecto en función de los objetivos planteados, identificando fortalezas y áreas de mejora.

Introducción

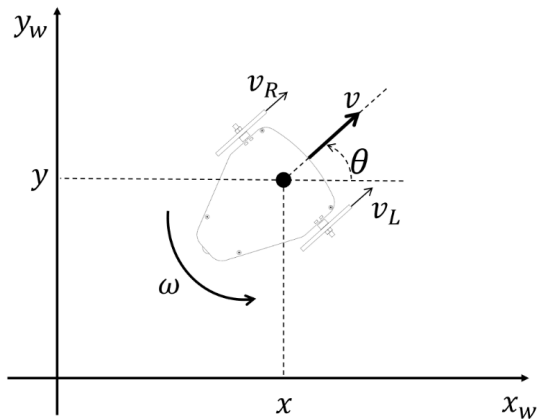
En la robótica móvil, uno de los principales desafíos es el control preciso del movimiento del robot en un entorno cambiante. El control en lazo cerrado es una técnica que se utiliza para mantener el movimiento del robot dentro de los parámetros deseados y corregir los errores de posición en tiempo real. En este sentido, el controlador proporcional integral derivativo (PID) es uno de los métodos más comúnmente utilizados en la implementación del control en lazo cerrado para robots móviles diferenciales.

El PID se basa en el cálculo del error entre la posición deseada del robot y su posición real, y ajusta los parámetros del controlador para minimizar este error. Sin embargo, existen varios temas derivados al cálculo del error y la robustez de un controlador que deben ser considerados en la implementación del PID para garantizar un control preciso y confiable del robot.

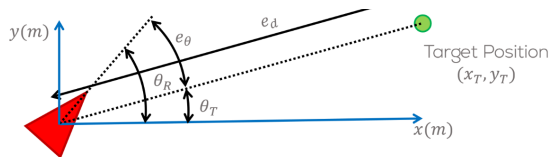
En este reporte de investigación, se abordará el tema del control en lazo cerrado para un robot móvil, enfocándose en la aplicación del PID para un robot móvil diferencial. Se discutirán los temas relacionados con el cálculo del error y la robustez de un controlador, así como las soluciones propuestas para abordar estos desafíos. El objetivo principal es proporcionar una visión general de los conceptos clave y los métodos utilizados en el control en lazo cerrado para robots móviles, y explorar los avances más recientes en el desarrollo de controladores robustos para mejorar el rendimiento de los robots móviles en diferentes entornos.

Solución del problema

En la primera parte de este reto, lo que realizamos fue obtener las medidas físicas de nuestro robot con la finalidad de poder definir el error lo que nos permitía calcular la posición tanto en X, Y y Z de nuestro robot.



Esto nos dio pie a poder calcular el error donde partimos de lo siguiente:



Como se puede apreciar existe una diferencia tanto en la distancia deseada y la distancia en la que se encuentra el objeto. Como tal podemos definir dos errores, el error angular y el error en la distancia que podemos obtener a través de distancias trigonométricas siendo la distancia la hipotenusa del triángulo formado por los dos puntos y el error angular estaría definido por el arcotangente del punto esperado y la resta del punto en el que se encuentra posicionado nuestro objeto, con esto procedimos a realizar un análisis en lazo abierto para poder visualizar el comportamiento de la planta. Una vez contando con más parámetros iniciamos la realización de nuestro controlador.

Lo que se realizó fue una simulación dentro de GAZEBO la cual nos ayudaría a poder desarrollar de una manera precisa y eficaz el controlador, siendo que para poder desarrollar este mismo utilizamos una serie de ecuaciones que hacían uso del error acumulado. Iniciamos trabajando con

el error en la distancia por lo que probamos un controlador P mediante método Heurístico pudimos realizar una mejora a la precisión de nuestro controlador, sin embargo no fue posible que el error tendiera a cero por lo que limitamos el dominio del error, es decir que al estar debajo de un umbral especificado el error pasaría como cero, posterior a esto realizamos el control para la velocidad angular siguiendonos bajo la misma lógica utilizamos un controlador PI que nos permitiría una precisión mayor al momento de la reducción del error de nuestro robot, así como el anterior se colocó un umbral el cual nos permitiera tener un mejor manejo del error.

Por otra parte diseñamos dos trayectorias para que nuestro robot se pudiera desplazar, cada una de estas se hizo con dos listas flotantes de parámetros los cuales tomarían el papel de X y Y e irían cambiando si el error angular y el error en la distancia fueran cero, es decir que hayamos llegado al punto deseado.

Para finalizar estructuramos nuestro entorno de trabajo de ROS utilizando mensajes, parámetros y clases las cuales nos permitieron trabajar de manera más limpia y eficiente, facilitando así la comunicación entre nuestro robot y nuestro entorno de trabajo.

Resultados

Los resultados obtenidos fueron el desplazamiento de dos trayectorias con cuatro puntos cada una, siendo que la primera emula la figura de un cuadrado mientras que la segunda emula un zig zag,

A continuación se puede visualizar el desplazamiento de las trayectorias.

[E3002B Intelligent Robotics Implementation](#)
[n](#)

Cuadrado: <https://youtu.be/2A7cIpV-CA8>

Zig zag: <https://youtu.be/WDWz-631YHU>

Conclusiones

Para la resolución de este reto se utilizaron dos controladores, un controlador PI para los ángulos y un controlador P para las distancias, esto se realizó de esta forma debido a que al utilizar dos medidas (Distancia y Ángulos), los errores que se generaban al trabajarlos juntos las necesidades de uno, no complementaban las necesidades del otro haciendo que la medición y la calibración no fueran las adecuadas, mientras que si separabamos ambos puntos podríamos calibrar de una mejor manera el desplazamiento de ambos, por otra parte pudimos apreciar que para poder mejorar la precisión de nuestro controlador es necesario el tomar en cuenta más entradas de sensores como lo son la cámara debido a que así podemos realizar de mejor manera un análisis del entorno del robot.

Referencias

ManchesterRoboticsLtd. (2023). GitHub - ManchesterRoboticsLtd/TE3002B_Intelligent_Robotics_Implementation. Retrieved 26 April 2023, from <https://github.com/ManchesterRoboticsLtd/T>