Tecnológico de Monterrey

Actividad R_2. Cálculo del error de un robot móvil diferencial

Carlos Adrián Delgado Vázquez A01735818

Alfredo Díaz López A01737250

Juan Paulo Salgado Arvizu A01737223

Bruno Manuel Zamora García A01798275

22 de abril del 2025

Resumen

En esta actividad se desarrollaron nodos en ROS2 orientados a la localización y navegación básica del robot móvil Puzzlebot. Se implementó un nodo para estimar la posición del robot utilizando datos de odometría obtenidos de los encoders, permitiendo conocer en todo momento sus coordenadas y orientación. Además, se creó un segundo nodo para calcular y publicar los errores de distancia *ed* y ángulo *e*θ respecto a una posición objetivo definida por el usuario.

Objetivos

Objetivo principal

Desarrollar e implementar nodos en ROS2 que permitan estimar la posición del Puzzlebot mediante odometría y calcular en tiempo real los errores de distancia y orientación hacia una posición objetivo, aplicando conceptos de localización y navegación punto a punto en lazo abierto.

Objetivos particulares

- Implementar un nodo para calcular la posición del robot a partir de los datos de los encoders.
- Verificar la estimación de posición utilizando el control manual mediante teleop_twist_keyboard.
- Crear un nodo para calcular y publicar los errores de distancia *ed* y orientación *e*θ hacia un objetivo.
- Observar en tiempo real la actualización de los errores al mover el robot en el entorno.
- Reforzar la comprensión práctica de los conceptos de odometría y navegación en lazo abierto.
-

Introducción

La localización es uno de los componentes fundamentales en la navegación de robots móviles, ya que permite conocer su posición y orientación dentro de un entorno. Esta práctica se enfoca en aplicar conceptos básicos de ubicación utilizando odometría en un robot móvil diferencial. La odometría se define como la técnica para medir las distancias recorridas por una persona, un vehículo o un autómata (Rae & Rae, n.d.). La odometría consiste en estimar la trayectoria recorrida por el robot a lo largo del tiempo. Esta se representa como una secuencia de poses consecutivas, es decir, posiciones y orientaciones, calculadas con respecto a una pose inicial que define el marco de referencia propio de la odometría. Es importante destacar que este sistema no se relaciona con un marco externo o global, como podría ser un mapa, sino que se basa únicamente en datos internos del robot. (*(7) ¿Qué Es La Odometría? | LinkedIn*, 2022).

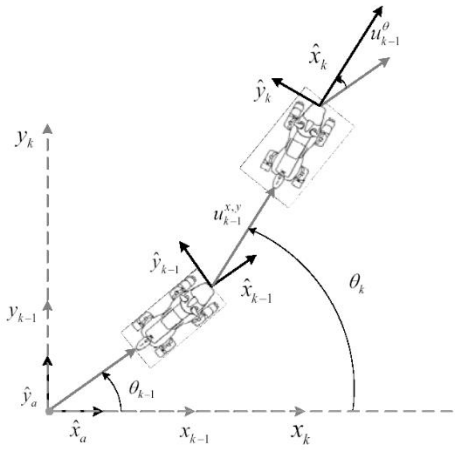

Ilustración 1 Movimiento basado en odometría.

La odometría, al no contar con una referencia global, tiende a desviarse con respecto a la trayectoria real recorrida por el robot. Esta desviación, se incrementa de forma continua con el tiempo. En sistemas de alta precisión, la acumulación de error puede limitarse a unos pocos centímetros y grados incluso tras trayectos largos (mayores a 1 km), mientras que en sistemas menos precisos la deriva puede alcanzar varios metros en recorridos de apenas cien metros. A pesar de esta limitación, los robots utilizan la odometría como una estimación inmediata de su posición, útil para apoyar procesos de localización dentro de un marco de referencia global, como un mapa. Estas estimaciones suelen ser válidas en escalas de tiempo muy cortas, que van desde unos pocos cientos de milisegundos hasta algunos segundos. (*(7) ¿Qué Es La Odometría? | LinkedIn*, 2022).

Un robot con tracción diferencial es un tipo de vehículo móvil que emplea dos ruedas motrices controladas de forma independiente, cada una impulsada por su propio motor. Su desplazamiento se basa en la diferencia de velocidad entre estas ruedas, las cuales se encuentran montadas sobre un mismo eje. Para mantener el equilibrio, se incorpora una tercera rueda auxiliar, conocida como castor, ubicada en la parte frontal o trasera del robot, la cual gira libremente. Cuando ambas ruedas giran en la misma dirección y a igual velocidad, el robot avanza en línea recta; si se invierte el sentido de giro manteniendo la misma velocidad, el robot retrocede. Por otro lado, si las ruedas giran a igual velocidad pero en sentidos opuestos, el robot rota sobre su propio eje, ya sea en sentido horario o antihorario. (Edisonsasig, 2023).

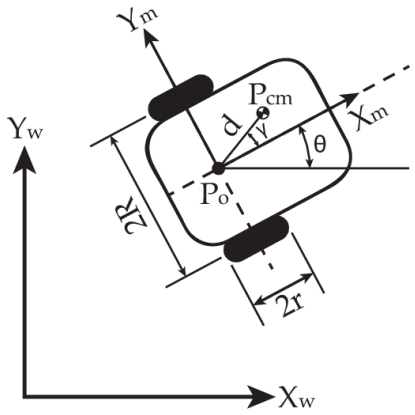

Ilustración 2 Robot movil diferencial

Para calcular el error de un robot móvil diferencial primero debemos definir qué es el error y en el contexto de los robots móviles el error es la diferencia entre la posición y orientación que el robot debería tener (posición deseada) y la que realmente tiene (posición real). Entonces el objetivo de aplicar técnicas de control es hacer que ese error tienda a cero.
Para obtener dicho error se calcula:

- Posición deseada y orientación deseada: $x_d,\ y_d, \theta_d$
- Posición real y la orientación real del robot: $x_r,\ y_r, \theta_r$

Lo primero que se hace es encontrar la diferencia entre las coordenadas deseadas y reales

$$\Delta x = x_d - x_r$$
$$\Delta y = y_d - y_r$$

Pero en robótica diferencial es más útil medir el error respecto al marco del robot, no respecto al marco global. Por eso, se aplica una rotación (transformación de coordenadas) que convierte el error al marco local del robot (Cortés et al., 2015):

$$\begin{bmatrix} x_e \\ y_e \end{bmatrix} = \begin{bmatrix} cos\theta_r & sin\theta_r \\ -sin\theta_r & cos\theta_r \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

En este sentido x_e es cuánto está el objetivo más adelante o atrás respecto al robot. Mientras que y_e es cuánto está el objetivo más a la izquierda o derecha.
Esto se hace porque el robot no se mueve directamente en x o y, sino que se mueve hacia adelante y gira sobre sí mismo y es más natural y efectivo diseñar el control desde el punto de vista del robot. Es decir, es más sencillo programar a un robot para que solo tenga que avanzar hacia adelante, atrás, izquierda o derecha en vez de avanzar hacia el norte 2 metros.

Además, el error de orientación es simplemente:

$$\theta_e = \theta_d - \theta_r$$

Entonces si el $\theta_e > 0$ el robot debe girar hacia la izquierda para tratar de minimizar este error pero en contraparte si $\theta_e < 0$ el robot debe girar hacia la derecha.

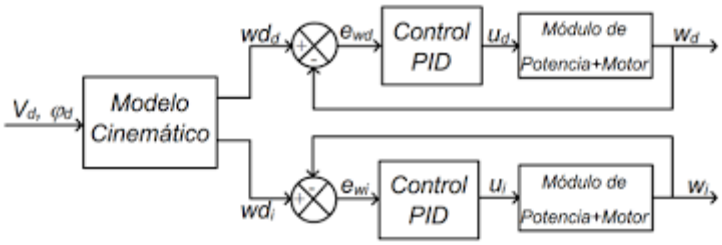

Ilustración 3Control de un robot móvil diferencial

Solución del problema

Metodología

1. Objetivo
    a. Desarrollar e integrar dos nodos en ROS 2:
        i. Localización: estimar y publicar la pose (x, y, θ) a partir de los encoders.
        ii. Cálculo de errores: recibir la pose estimada y la meta, calcular y publicar e_d (distancia) y e_θ (ángulo).
        iii.
2. Diseño de la solución
    a. Nodo Localización
        i. Lee periódicamente las velocidades de rueda.
        ii. Calcula la pose acumulada mediante integración discreta.
        iii. Publica la odometría estimada.
    b. Nodo Errores
        i. Escucha la odometría y la posición objetivo.
        ii. Calcula e_d = distancia(actual, meta) y e_θ = ángulo(normalizado).
        iii. Publica ambos errores en sus respectivos tópicos.

3. Parámetros y configuración
    a. Parámetros físicos: distancia entre ruedas y radio, declarados externamente
    b. Frecuencia de muestreo: elegir un intervalo estable.
    c. QoS: ajustar fiabilidad y latencia para odometría y errores según necesidad.

4. Procedimiento experimental
    a. Definir parámetros en archivo de configuración o al lanzar los nodos.
    b. Iniciar el nodo de localización.
    c. Ejecutar teleop_twist_keyboard para mover el robot manualmente.
    d. Iniciar el nodo de errores y publicar la meta (ros2 topic pub /pose …).
    e. Monitorear los tópicos de odometría y errores con ros2 topic echo y/o rqt_plot.

5. Verificación de resultados
    a. Comprobar que la pose (x, y) varía coherentemente con el movimiento.
    b. Verificar el valor e_d.
    c. Asegurar que e_θ se mantiene dentro de un margen sin saltos bruscos.
6. Iteración y ajuste fino
    a. Si la estimación “salta” o se retrasa, ajustar el intervalo de muestreo.
    b. Corregir los parámetros físicos si hay sesgos.
    c. Refinar la normalización de ángulos para eliminar discontinuidades.

Resultados

Los resultados que obtuvimos para solucionar esta actividad se presentan a continuación, las capturas de pantalla representan los datos que obtuvimos en la publicación de la edometría y
Se hicieron pruebas en un principio para avanzar 3 metros, de los cuales vimos que al avanzar 2.8 metros, en realidad, completó los 3 metros, como se ven en la Ilustración 4 y Ilustración 5, donde esta misma, calcula el error con respecto a la distancia real.

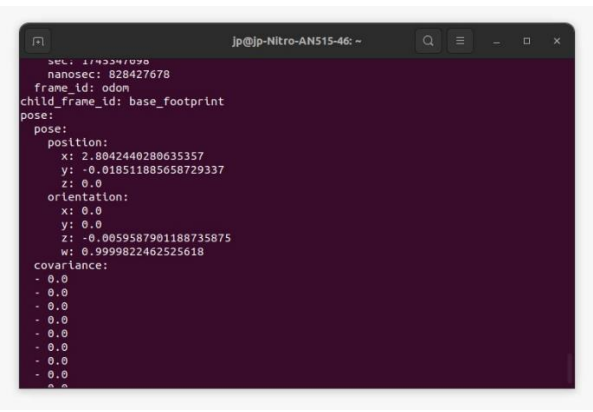

Ilustración 4 Odometría del puzzlebot a 3 metros reales

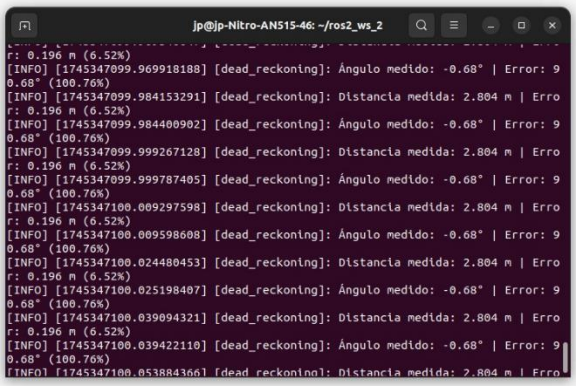

Ilustración 5 Cálculo del error a 3 metros reales

Posteriormente, se realizó el mismo cálculo pero para 5 metros y se obtuvo los resultados como se ve en la imagen Ilustración 6 y Ilustración 67.

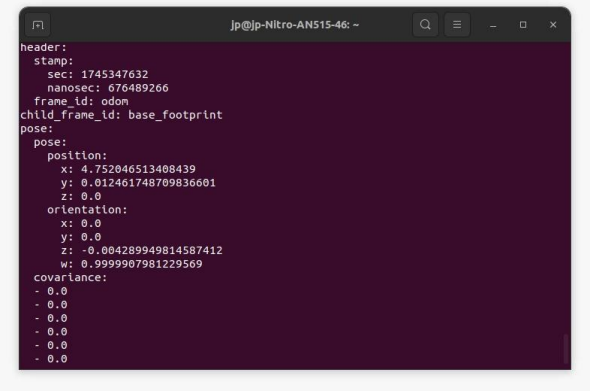

Ilustración 6 Odometría del puzzlebot a 5 metros reales

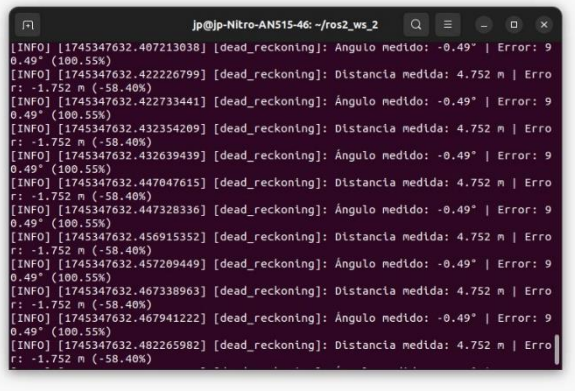

Ilustración 7 Cálculo del error a metros reales

Se presentan los videos demostrativos del funcionamiento de la odometría y el cálculo de errores de distancia lineal y de giro en el siguiente link de Google drive:
https://drive.google.com/drive/folders/1SIHdR5-InjVHua-Bd-Rrsko1h3_yME3K?usp=sharing

Conclusiones

Se lograron cumplir los objetivos establecidos. El nodo para calcular la localización del robot mediante odometría funcionó correctamente, y el cálculo de los errores de distancia y orientación se actualizó como se esperaba. No obstante, la precisión de la estimación de la posición se vio

afectada por la acumulación de errores en la odometría, lo cual es un comportamiento esperado debido a la naturaleza del sistema.
Una posible mejora sería integrar sensores adicionales, como las cámaras, para reducir la deriva y mejorar la precisión. A pesar de estas limitaciones, los resultados obtenidos contribuyen a una comprensión práctica de la localización y la navegación en lazo abierto.

Bibliografía o referencias

Rae, & Rae. (n.d.). *odometría | Diccionario histórico de la lengua española*. «Diccionario Histórico De La Lengua Española». https://www.rae.es/dhle/odometr%C3%ADa

(7) ¿Qué es la odometría? | LinkedIn. (2022, September 19). https://www.linkedin.com/pulse/qu%C3%A9-es-la-odometr%C3%ADa-arme-rob%C3%B3tica-m%C3%B3vil/

Edisonsasig. (2023, July 14). *Modelo cinemático y simulación de un robot móvil diferencial*. Roboticoss. https://roboticoss.com/modelo-cinematico-y-simulacion-con-python-robot-movil-diferencial/

Cortés, U., Castañeda, A., Benítez, A., & Díaz, A. (2015, octubre). *Control de movimiento de un robot móvil tipo diferencial Robot ůßot-32b*. Congreso Nacional de Control Automático, AMCA 2015, Cuernavaca, Morelos, México.