Proyecto 2: Control Cinemático de Robots-Generación

De Trayectorias en ROS

Robótica SDI-11911;

Instituto Tecnológico Autónomo de México

Departamento Académico de Sistemas Digitales

Otoño de 2016

Diego Pozo, Alan Pérez, Erick Zetina, Ruiciro Rivera

**Introducción:**

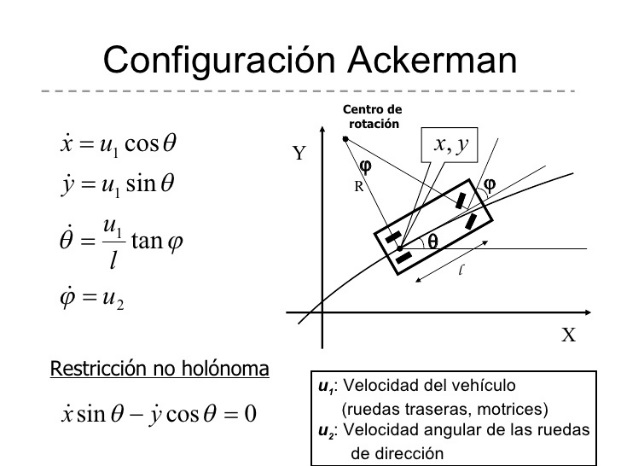
***El control cinemático es fundamental al momento del diseño de robots, ya que este se encarga, una vez obtenido los datos necesarios (punto de destino, tiempo, error asociado) en generar una trayectoria que se ajuste al modelo del motor, para posteriormente generar un muestreo que sirva de referencia al control dinámico.***

**Marco Teórico:**

**Gazebo:** Ofrece la posibilidad de crear poblaciones de robots y simularlos con alta precisión sin importar si son ambientes cerrados o al aire libre.

**Modelo ACKERMAN**: Es comúnmente utilizado en robots de cuatro ruedas, el sistema se basa en dos ruedas traseras tractoras montadas paralelamente en el chasis del vehículo. Mientras que las ruedas delanteras se utilizan para direccionamiento y son controladas por un motor independiente de las ruedas traseras. Una limitante del modelo Ackerman es la

maniobrabilidad, ya que al ejecutarse la acción de giro,  la rueda interna, gira con un ángulo ligeramente mayor a la rueda exterior.



**Navegación de robots vía sensores:**

El sistema de navegación en los robots autónomos es una de las partes más importantes, pues es quien le permite al robot posicionarse respecto al mundo y poder crear trayectorias, a pesar de que hay muchas maneras en la que un robot puede obtener datos del mundo exterior, el uso de sensores resulta muy práctico, ya que además de ser económico, son adaptables y la información obtenida suele ser muy precisa.

Artículos revisados: “Sensores, métodos y técnicas para la navegación de robots móviles Autónomos”

**Descripción de la solución**

Se utilizó el modelo Autonomus con las siguientes características físicas:

Chasis: .144m de ancho, .445m de largo, .080m de ancho.

Rueda: .31m de radio, .45m de largo.

**Algoritmos de control usados para cada problema:**

**Lane keeping:** Una vez obtenida las posición actual las posición actual del coche, se calculaba la distancian entre puntos que eran proporcionados por la nube de datos que nos arrojaba el nodo visión, una vez obtenida estos datos, se transformaban los datos de manera que el robot pudiera entenderlos desde su propio mundo, pues los datos arrojados de la nube estaban en función del mundo no del robot ,una vez transformados los datos se sacaba un promedio de estos para lograr que el auto fuera en medio del carril.

Un problema que se enfrento fue, la corrección del error asociado, pues este se iba acumulando y se hacía más grande mientras más tiempo transcurría.

*Algoritmo Lane keeping:*

* Obtención de datos a partir del nodo visión
* Transformación de los datos obtenidos al mundo del robot
* Una vez obtenidos los datos en “mundo robot” se calculo el esfuerzo necesario para motor
* Sacar un promedio de los datos obtenidos
* Aplicar el esfuerzo necesario para corregir y mantenerse en el carril deseado.

**Experimentos**

Como primer experimento, nos dimos a la tarea de descubrir cómo funcionan los archivos y aprender a usar los comandos de roslaunch y ros run, posteriormente a esto, sé uso como base el modelo del Ekbot, y enfrentamos complicaciones para lograr una comunicación exitosa entre nodos.

Una vez que se superó el reto de la comunicación entre nodos, nos enfocamos en el modelo matemático, para lograr transformar los datos obtenidos, en que el robot se moviera en una trayectoria previamente creada.

Tras algunas modificaciones basadas en prueba y error se pudo encontrar la constante necesaria para que el robot tuviera la velocidad deseada, que no fuera muy lento pero tampoco saliera volando.

El siguiente reto fue encontrar una manera de corregir el error asociado, para esto se planteó el uso de la función flush o corregir el modelo matemático del coche.

**Conclusiones**

Gracias a este proyecto, se pudo apreciar las problemáticas que uno como ingeniero enfrenta, en la construcción y diseño de programas y algoritmos, que sirvan para realizar el control de robots autónomos, así como la construcción de sus trayectorias. Y la importancia del error asociado y como corregirlo.

**Referencias**

http://www.geocities.ws/robottotem/Download/Sensores.pdf