

Nombre: Hernández Peñaafort Marco Antonio

**1. Explique qué es la configuración, espacio de configuraciones y grados de libertad de un robot móvil.**

La configuración de un robot describe completamente la posición de algún punto del mismo. Por otro lado, el espacio de configuraciones se refiere al espacio de todas las configuraciones posibles del sistema, es decir, todas las posiciones que puede tomar el robot. Por lo tanto, una configuración es simplemente un punto en el espacio de configuraciones, por ejemplo  $(x, y, z, \theta)$ . El número de grados de libertad de un robot móvil se refiere a la dimensión del espacio de configuraciones, o el número mínimo de parámetros necesarios para especificar la configuración.

**2. Investigue dos métodos basados en grafos para planeación de rutas.**

Algoritmo de Dijkstra: es un algoritmo de búsqueda de grafos que encuentra la ruta más corta de una sola fuente en el grafo. Suponga que cada borde,  $e \in E$ , en la representación por grafos de un problema de planificación discreta tiene un costo no negativo asociado  $l(e)$ , que es el costo para aplicar la acción. El costo  $l(e)$  podría escribirse usando la representación del espacio de estado como  $l(x, u)$ , lo que indica que cuesta  $l(x, u)$  aplicar la acción  $u$  desde el estado  $x$ . El costo total de un plan es solo la suma de los costos de borde sobre la ruta desde el estado inicial a un estado objetivo.

El algoritmo de búsqueda A-star ( $A^*$ ) es una extensión del algoritmo de Dijkstra que intenta reducir el número total de estados explorados mediante la incorporación de una estimación heurística del costo para llegar a la meta desde un estado determinado. Su aspecto de diseño más importante es la determinación de la función de costo, que define los pesos de los nodos. Es adecuado para buscar espacios conocidos principalmente *a priori* por el vehículo, pero costosos en términos de memoria y velocidad para vastas áreas.

**3. Investigue dos métodos basados en muestreo para planeación de rutas.**

Estos planificadores intentan resolver las restricciones de tiempo, es decir, la planificación en espacios de alta dimensión, que los métodos deterministas no pueden cumplir. El enfoque consiste en muestrear aleatoriamente el espacio de configuración o espacio de estado, buscando conectividad dentro de él. La desventaja es el hecho de que la solución es subóptima. Los métodos más utilizados en robótica son el Método de mapas probabilísticos (PRM) y el Árbol aleatorio de exploración rápida (RRT).

Árbol aleatorio de exploración rápida (RRT): pertenece a los algoritmos basados en muestreo aplicables a la planificación de rutas en línea. Permite una planificación rápida en espacios semiestructurados mediante la ejecución de una búsqueda aleatoria a través del área de navegación. También tiene la capacidad de considerar restricciones no holonómicas (como el radio de giro máximo y el momento del vehículo).

Por otro lado, la metodología general de los PRM es construir un gráfico (el mapa) durante el preprocesamiento que representa la conectividad del espacio de configuraciones libres del robot y luego consultar el mapa para encontrar una ruta para una tarea de planificación de movimiento determinada.

**4. Explique en qué consiste el proceso de SLAM (Simultaneous Localization and Mapping).**

SLAM es un proceso mediante el cual un robot móvil puede construir un mapa de un entorno y, al mismo tiempo, usar este mapa para deducir su ubicación. En SLAM, tanto la trayectoria de la plataforma como la ubicación de todos los puntos de referencia se estiman linealmente sin la necesidad de ningún conocimiento *a priori* de la ubicación.

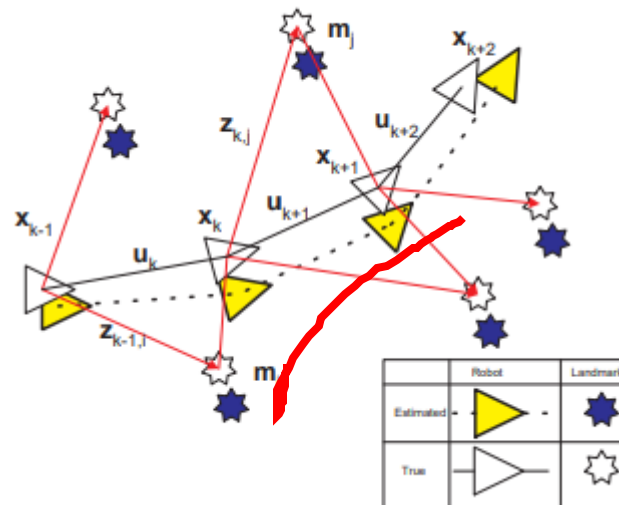
Considere un robot móvil que se mueve a través de un entorno tomando observaciones relativas de

varios puntos de referencia desconocidos utilizando un sensor ubicado en el robot. En un instante instantáneo  $k$ , se definen las siguientes cantidades:

- $x_k$ : el vector de estado que describe la ubicación y orientación del vehículo.
- $u_k$ : el vector de control, aplicado en el tiempo  $k$  para conducir el vehículo a un estado  $x_k$  en el tiempo  $k$ .
- $m_i$ : Un vector que describe la ubicación del  $i$ -ésimo punto de referencia cuya ubicación real se supone invariante en el tiempo.
- $z_{ik}$ : Una observación tomada desde el vehículo de la ubicación del  $i$ -ésimo punto de referencia en el tiempo  $k$ . Cuando hay múltiples observaciones de puntos de referencia en cualquier momento o cuando el punto de referencia específico no es relevante para la discusión, la observación se escribirá simplemente como  $z_k$ .

Además, también se definen los siguientes conjuntos:

- $X_{0:k} = \{x_0, x_1, \dots, x_k\} = \{X_{0:k-1}, x_k\}$ : el historial de ubicaciones de vehículos.
- $U_{0:k} = \{u_1, u_2, \dots, u_k\} = \{U_{0:k-1}, u_k\}$ : El historial de entradas de control.
- $m = \{m_1, m_2, \dots, m_n\}$ : El conjunto de todos los puntos de referencia.
- $Z_{0:k} = \{z_1, z_2, \dots, z_k\} = \{Z_{0:k-1}, z_k\}$ : el conjunto de todas las observaciones de puntos de referencia.



El problema esencial de SLAM. Se requiere una estimación simultánea de las ubicaciones de robot y de los puntos de referencia (*landmarks*). Las ubicaciones verdaderas nunca se conocen ni se miden directamente. Las observaciones se hacen entre el verdadero robot y los puntos de referencia.

## 5. Explique en qué consiste la localización mediante filtros de partículas, sus ventajas sobre el Filtro de Kalman y los paquetes de ROS que lo implementan.

Dado que los filtros de Kalman usan distribuciones gaussianas, no pueden representar adecuadamente las creencias que corresponden a situaciones ambiguas tal como aparecen, por ejemplo, en el contexto de la localización global. Como resultado, los enfoques de localización que usan filtros Kalman generalmente requieren que se conozca la ubicación inicial del robot o que se den puntos de referencia únicos para que no haya ningún problema de asociación de datos.

Una forma alternativa y eficiente de representar y mantener las densidades de probabilidad es el filtro de partículas. La idea clave de los filtros de partículas es representar la parte posterior mediante un conjunto  $M$  de  $N$  muestras. Cada muestra consta de un par  $(x, \omega)$  que contiene un vector de estado  $x$  del sistema subyacente y un factor de ponderación  $\omega$ , es decir,  $M = (X, [0, 1])^N$ . Este último se utiliza para almacenar la importancia de la partícula correspondiente. El posterior está representado por la distribución de las muestras y sus factores de importancia.

Para implementarlo en ros, se requiere de la librería *bfl* (*Bayesian Filtering Library*). Algunos paquetes de ROS que lo implementan son: *mcl\_pi* & *amcl*.

El paquete *mcl\_pi* se basa en el uso de una técnica de fusión distribuida para el filtro de partículas

denominada "Intersección de partículas". Este paquete contiene dos clases principales: implementación de localización de Monte Carlo (MCL) en python, y una intersección de partículas (PI).

El paquete amcl es un sistema de localización probabilístico para un robot que se mueve en 2D. Implementa el enfoque adaptativo (o muestreo KLD) de localización de Monte Carlo, que utiliza un filtro de partículas para rastrear la pose de un robot contra un mapa conocido.

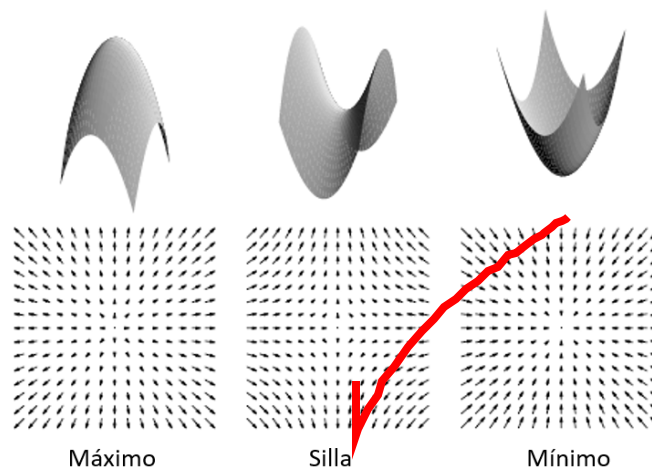
## 6. Investigue qué son los campos potenciales y explique los pasos generales para implementarlos.

Una función potencial (o campo potencial) es una función de valor real diferenciable  $U: \mathbb{R}^m \rightarrow \mathbb{R}$ . El valor de una función potencial puede verse como energía y, por lo tanto, el gradiente del potencial es la fuerza. El gradiente es un vector  $\nabla U(q) = DU(q)^T = [\frac{\partial U}{\partial q_1}(q), \dots, \frac{\partial U}{\partial q_m}(q)]^T$  que apunta en la dirección que localmente aumenta al máximo  $U$ .

Usamos el gradiente para definir un campo vectorial, que asigna un vector a cada punto en una variedad. Un campo vectorial de gradiente, como su nombre lo indica, asigna el gradiente de alguna función a cada punto. Cuando  $U$  es energía, el campo del vector gradiente tiene la propiedad de que el trabajo realizado a lo largo de cualquier camino cerrado es cero.

El enfoque de función potencial dirige a un robot como si fuera una partícula que se mueve en un campo vectorial de gradiente. Los gradientes se pueden ver intuitivamente como fuerzas que actúan sobre un robot de partículas con carga positiva que se siente atraído por el objetivo con carga negativa. Los obstáculos también tienen una carga positiva que forma una fuerza repulsiva que aleja al robot de los obstáculos. Con suerte, la combinación de fuerzas repulsivas y atractivas dirige al robot desde la ubicación inicial a la ubicación objetivo, evitando obstáculos.

En este enfoque, vemos los gradientes como vectores de velocidad en lugar de vectores de fuerza. Las funciones potenciales pueden verse como un paisaje donde los robots se mueven de un estado de "alto valor" a un estado de "bajo valor". El robot sigue un camino "cuesta abajo" siguiendo el gradiente negativo de la función potencial. Seguir una ruta de este tipo se denomina **descenso en gradiente**, es decir,  $\dot{c}(t) = -\nabla U(c(t))$ . El robot termina el movimiento cuando alcanza un punto donde el gradiente se desvanece, es decir, ha alcanzado a  $q^*$  donde  $\nabla U(q^*) = 0$ . Tal punto  $q^*$  se llama un punto crítico de  $U$ . El punto  $q$  puede ser ya sea un punto máximo, mínimo o punto silla.



## 7. Explique qué es una transformación homogénea y para qué se utiliza en robots móviles.

Una transformación homogénea se refiere a la transformación de un vector de coordenadas homogéneas de un sistema de coordenadas a otro. Esta se representa mediante las llamadas matrices de transformación homogéneas.

Las matrices de transformación homogénea se utilizan para:

- Representar la posición y orientación de un sistema girado y trasladado con respecto a un sistema fijo.

- Transformar un vector expresado en coordenadas movibles y su representación en un sistema fijo.
- Rotar y trasladar un vector con respecto a un sistema fijo.

Una matriz de transformación homogénea  $T$  es una matriz de dimensión  $4 \times 4$  que se haya compuesta por 4 sub-matrices:

- $R_{3 \times 3}$ , matriz de rotación.
- $p_{3 \times 1}$ , vector de traslación.
- $f_{1 \times 3}$ , transformación de perspectiva.
- $w_{1,1}$ , factor de escala.

$$T = \begin{bmatrix} R_{3 \times 3} & p_{3 \times 1} \\ f_{1 \times 3} & w_{1 \times 1} \end{bmatrix} = \begin{bmatrix} \text{Rotación} & \text{Traslación} \\ \text{Perspectiva} & \text{Escala} \end{bmatrix}$$

## 8. Investigue qué es un robot con restricciones no holonómicas de movimiento.

Cada sistema robótico está sujeto a una variedad de restricciones de movimiento, pero no todas pueden expresarse como restricciones de configuración. Un ejemplo familiar de tal sistema es un automóvil. A bajas velocidades, las ruedas traseras del automóvil ruedan libremente en la dirección que apuntan, pero evitan el deslizamiento en la dirección perpendicular. Esta restricción implica que el automóvil no puede trasladarse directamente a un lado. Sin embargo, sabemos por experiencia que esta restricción de velocidad no implica una restricción en las configuraciones; El automóvil puede alcanzar cualquier posición y orientación en el plano libre de obstáculos. De hecho, la translación lateral antes descrita se puede aproximar mediante maniobras de estacionamiento en paralelo.

Esta restricción antideslizante es una restricción no holonómica, una restricción en la velocidad. Además de rodar sin resbalar, la conservación del momento angular es una fuente común de restricciones no holonómicas en los sistemas mecánicos.

### Fuentes consultadas:

- González, D., Pérez, J., Milanés, V., & Nashashibi, F. (2015). A review of motion planning techniques for automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 17(4), 1135-1145.
- Paden, B., Čáp, M., Yong, S. Z., Yershov, D., & Frazzoli, E. (2016). A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on intelligent vehicles*, 1(1), 33-55.
- Dissanayake, M. G., Newman, P., Clark, S., Durrant-Whyte, H. F., & Csorba, M. (2001). A solution to the simultaneous localization and map building (SLAM) problem. *IEEE Transactions on robotics and automation*, 17(3), 229-241.
- Choset, H. M., Hutchinson, S., Lynch, K. M., Kantor, G., Burgard, W., Kavraki, L. E., & Thrun, S. (2005). *Principles of robot motion: theory, algorithms, and implementation*. MIT press.
- H. Durrant-Whyte and T. Bailey, "Simultaneous localisation and mapping (SLAM): Part I, the essential algorithms," *IEEE Robotics and Automation Magazine*, vol. 13, no. 2, pp. 99–110, Jun. 2006.
- LaValle, S. M. 2006. *Planning Algorithms*. Cambridge, UK: Cambridge University Press.