



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN
IIC-2685 ROBÓTICA MÓVIL

LAB 4: NAVEGACIÓN

Fecha de Presentación: Miércoles 20 de Junio de 2018 14:00

Introducción

Vamos acercándonos al término del curso, y esta vez daremos libertad de movimiento a nuestros TurtleBots, dotándolos de una importante destreza: navegar por un ambiente conocido. Para esto usaremos todo el conocimiento adquirido durante los laboratorios pasados, es decir, usaremos la localización (Lab 3), necesaria para que el robot sepa dónde se encuentra dentro de un mapa, para luego planear una ruta óptima a seguir mediante movimientos programados (Lab 2) con feedback de la odometría y del ambiente.

Usaremos un mapa similar al del Laboratorio 3, aunque éste se entregará sólo el día de la entrega del laboratorio. Para sus pruebas pueden usar el mismo que el del Laboratorio 3.

Navegación

Para efectuar la navegación, a su programa se le deberá entregar dos poses (x, y, θ) , donde las unidades de x e y son metros y θ en radianes. Una pose corresponderá a una estimación de la pose inicial, y la segunda a una pose *goal*. Pueden suponer que siempre se entregarán poses dentro del mapa. Para simplificar la tarea, puede asumir que se entregará una pose inicial aproximada del robot, con un error de distancia de a lo más 0.3 metros y un error de orientación de a lo más 0.5 radianes. El laboratorio se compone de cuatro partes:

1. Localizar el robot dentro del mapa, dándole una pose inicial. Debe crear un nodo que suscriba un mensaje en un tópico llamado `/initialpose`, donde se le entregue una estimación de pose inicial. Una alternativa es que el tipo de mensaje sea `geometry_msgs/Pose`, aunque son libres de elegir cómo le entregan la pose inicial (por ejemplo, puede ser un String y formatear la pose como un JSON). Para localizar al robot, pueden usar los algoritmos de localización que programaron para el Laboratorio 3, usando partículas cercanas a la pose inicial definida. Una vez recibida la pose, el robot deberá efectuar una localización sin moverse del lugar inicial.
2. El segundo paso es recibir la pose deseada para la navegación. Deben seguir pasos similares para la pose inicial, pero esta vez usando un tópico llamado `/goal`. Con el *goal* definido, deben planear una ruta entre la pose actual y la pose *goal*. Para esto pueden usar un algoritmo de búsqueda que ustedes definan, aunque se recomienda usar A*, usando los *cells* del mapa como estados.
3. Con la ruta definida, es hora de mover al robot por la ruta seleccionada. Para esto, puede usar *sampling* de velocidad lineal y angular y tomar la más cercana al path en cada iteración. Otra alternativa simple es definir *waypoints* dentro de la ruta y mover al robot a cada *waypoint* en forma independiente (como lo pedido en el Lab 2), y chequear la localización en cada *waypoint*.

Para la realización del laboratorio, supondremos que no habrá obstáculos dentro del mapa, y que los archivos del mapa (`map.pgm` y `map.yaml`) corresponden a la realidad, es decir, es un mapa estático y actualizado.

Para la presentación, deben especificar cómo programaron el paso 2 y 3, qué dificultades encontraron y cómo las resolvieron.

1 Demostración

Para la demostración, a cada grupo se le pasará el mismo mapa el día de la presentación, y se les dará una pose inicial distinta. Deben disponer de los programas para publicar la pose inicial y el *goal* a los robots de forma que efectúen la navegación. El ayudante pondrá al robot en una pose aproximada a la pose inicial especificada. El éxito se logrará cuando el robot llegue al *goal* (o al menos cerca 20 cm del *goal*). No se penalizará el tiempo en que demore llegar. Una vez que llegue al goal, el robot debe avisar por sus parlantes que ha llegado.