

Memoria Práctica 2

Explorador basado en fronteras

Antonio Manuel Milán Jiménez Grupo 3

Inicio:

Inicialmente el robot realiza 2 giros de 360° para que actualice el entorno a su alrededor, es decir, lo que anteriormente le era desconocido y ahora lo conoce. El siguiente paso es que determine la frontera para saber cual será su siguiente movimiento.

Determinar la frontera:

Para determinar la frontera, se recorre todas las posiciones de la matriz “cmGlobal”. Por cada posición comprueba si es conocida y si alguna de las posiciones de su alrededor es desconocida, es decir, la definición de que un punto sea frontera. Cuando un punto cumple esta condición se añade a la frontera.

Como comprobación adicional, al terminar este proceso comprueba si la frontera esta vacía, lo cual significará que termina la exploración.

Elegir un nodo de la frontera:

Para elegir un nodo de la frontera se contempla 2 casos.

El primer caso es justo al inicio de la exploración y cuando se ha alcanzado el nodo objetivo en la anterior exploración. En este caso, recorre todo el vector de la frontera y calcula los 3 puntos de la frontera más cercanos. A continuación, se elige aleatoriamente entre los 3 puntos para determinar ya a cual dirigirse. El hecho de que se esté haciendo una elección aleatoria impide que el robot entre en un bucle intentando ir a un punto.

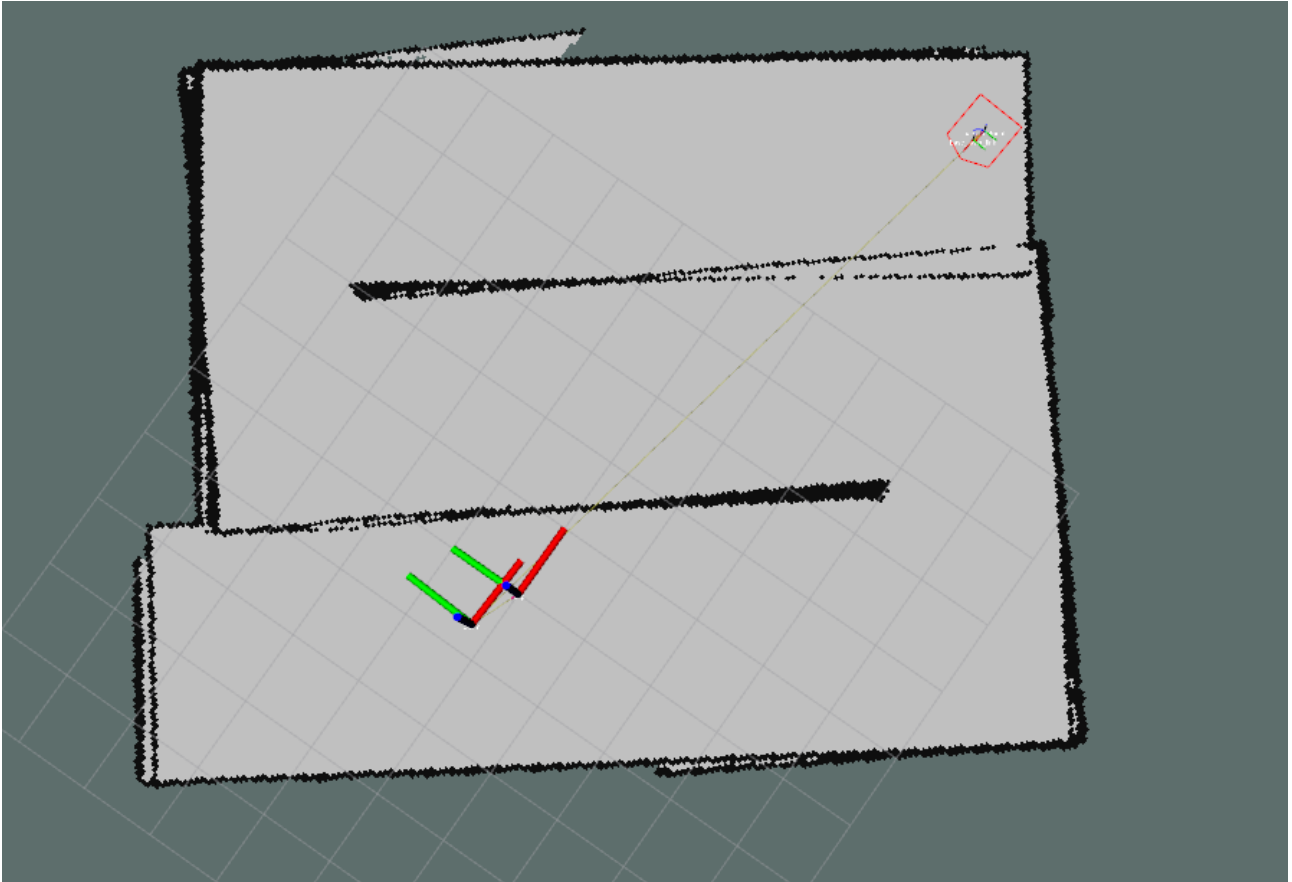
Fallo en alcanzar el nodo objetivo:

El segundo caso es cuando no ha conseguido alcanzar el nodo objetivo anterior por cualquier razón. En este caso, se elige un punto aleatorio de la frontera para evitar que nuevamente el robot intente ir a los puntos más cercanos ya que en ese momento pueden no ser accesibles.

Una vez que tiene seleccionado el nodo objetivo, se dirigirá a él mediante el método “moverRobot” que implementa el código de “SendGoals.cpp”.

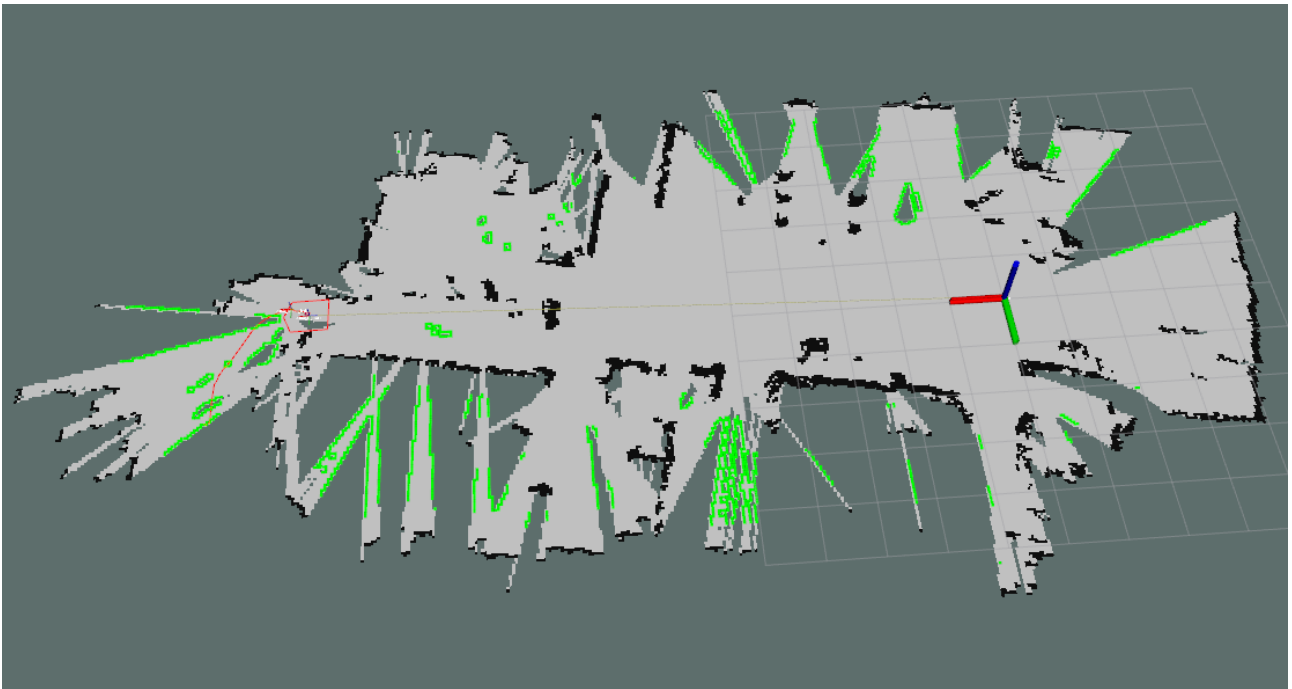
Experimentación

Para el mapa de “corridor”, el robot ha conseguido explorarlo por completo en 13 minutos. Además, dada la simplicidad del mapa, no ha tenido dificultades para hacerlo ni se chocado en ningún momento. Aquí podemos ver el mapa ya explorado:



Sin embargo, en el mapa de “willow garage”, dada la dificultad, el robot apenas consigue explorar la zona en la que empieza puesto que fácilmente acaba chocando y estancándose en una zona de la que ya no puede salir. También, al ser tan irregular el mapa, su ritmo de exploración es mucho más lento que en el “corridor”.

Aquí tenemos una de las ejecuciones donde ha explorado esta porción del mapa en 7 minutos:



Hasta que finalmente se ha quedado atascado en un pasillo de donde ya no podía salir:

