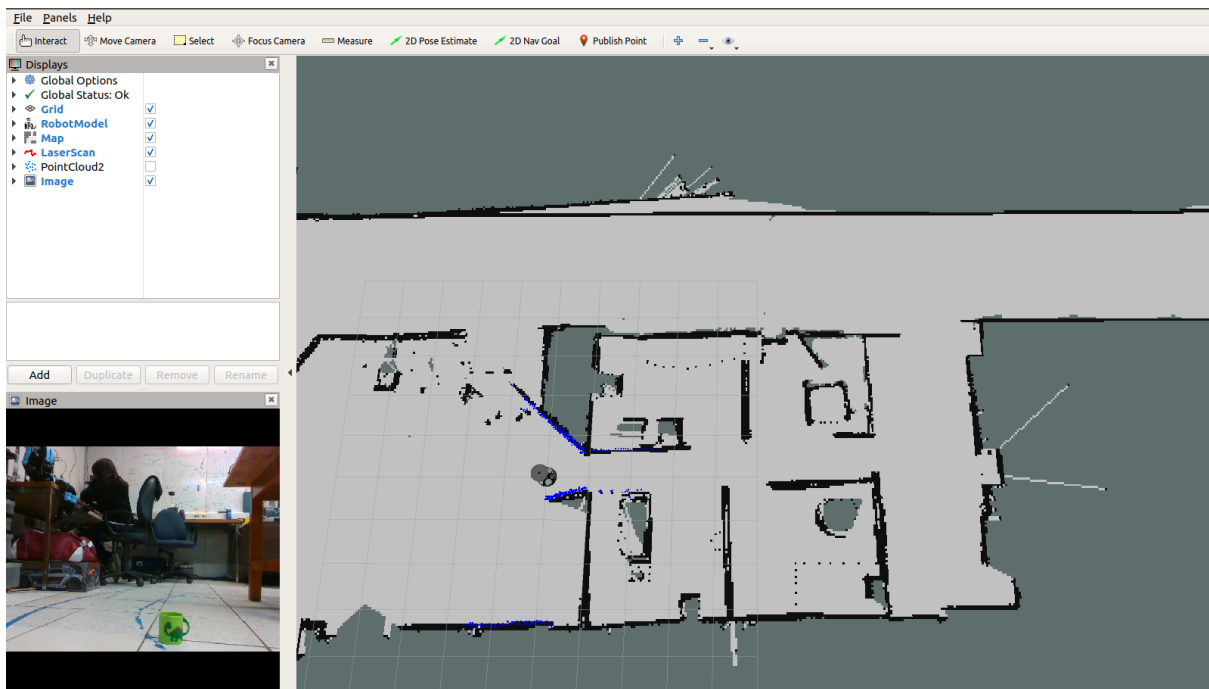


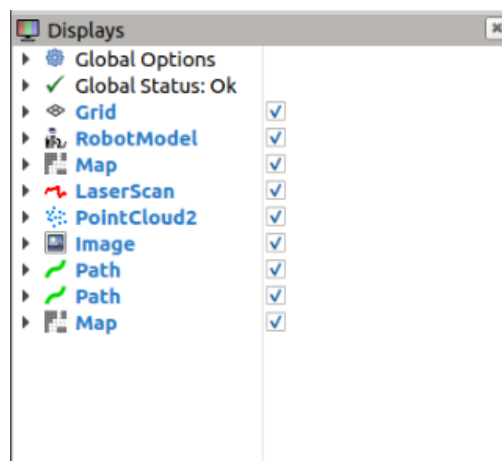
PRÁCTICA 3: USO DEL NAVIGATION STACK PARA NAVEGACIÓN EN 2D

DESARROLLO

Al principio de la práctica se tenía el siguiente mapa:

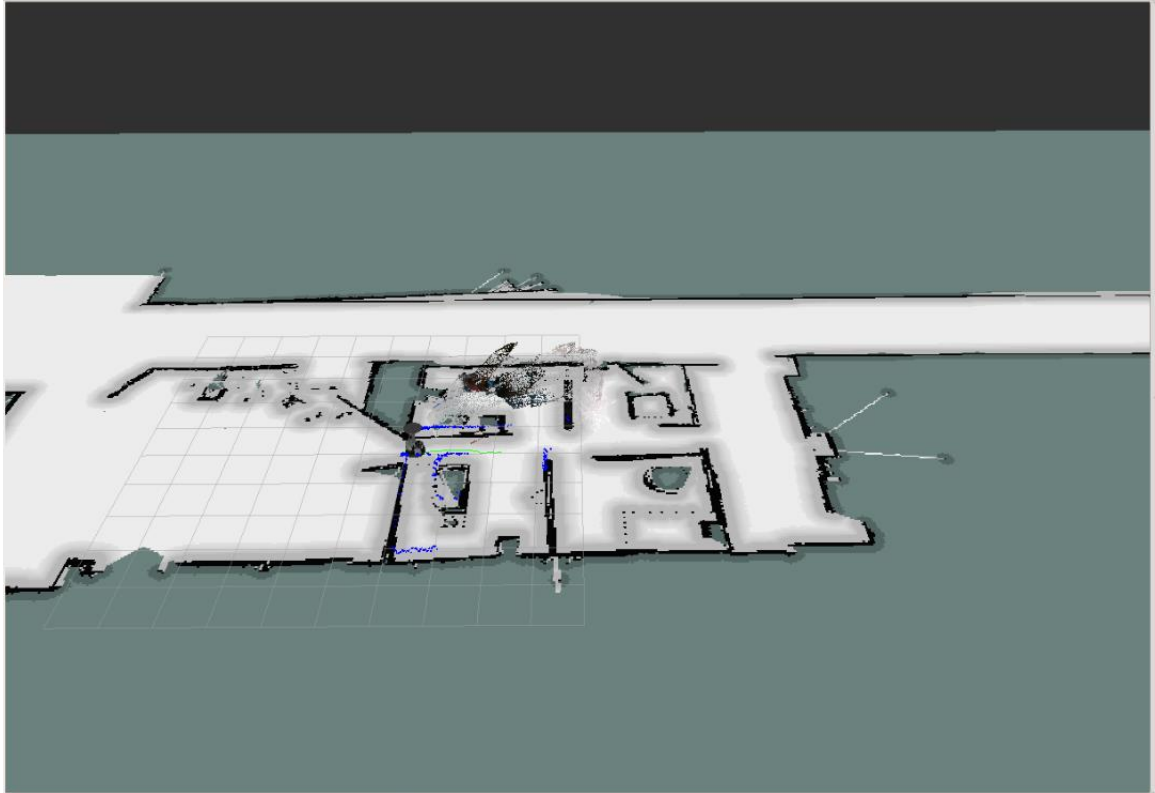


Posteriormente se procedió a ejecutar el `navigation_move_base.launch` y añadir los distintos display requeridos para la navegación.

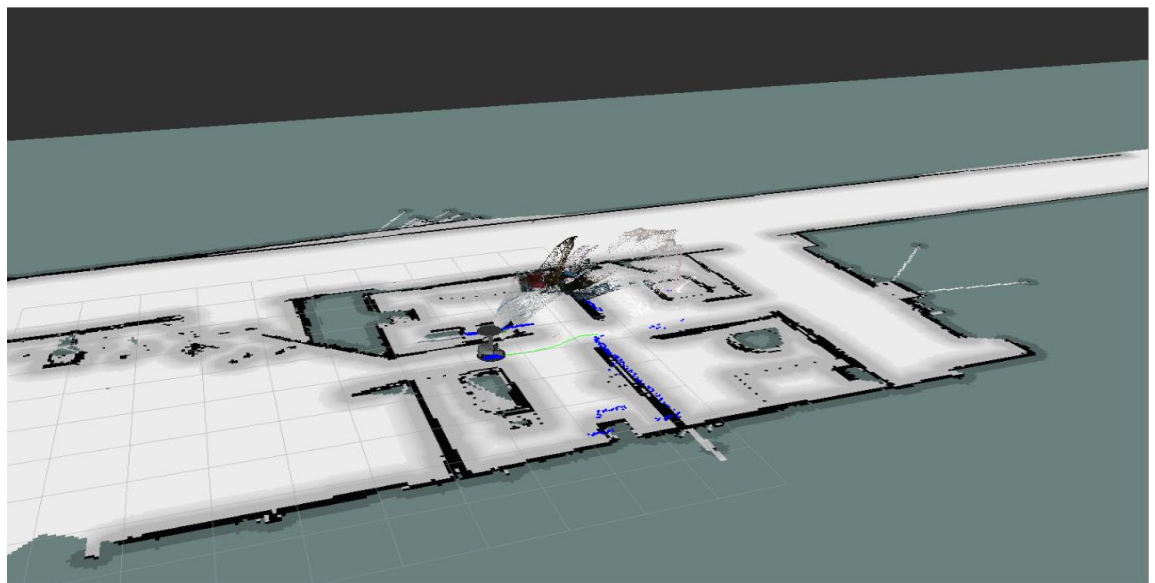


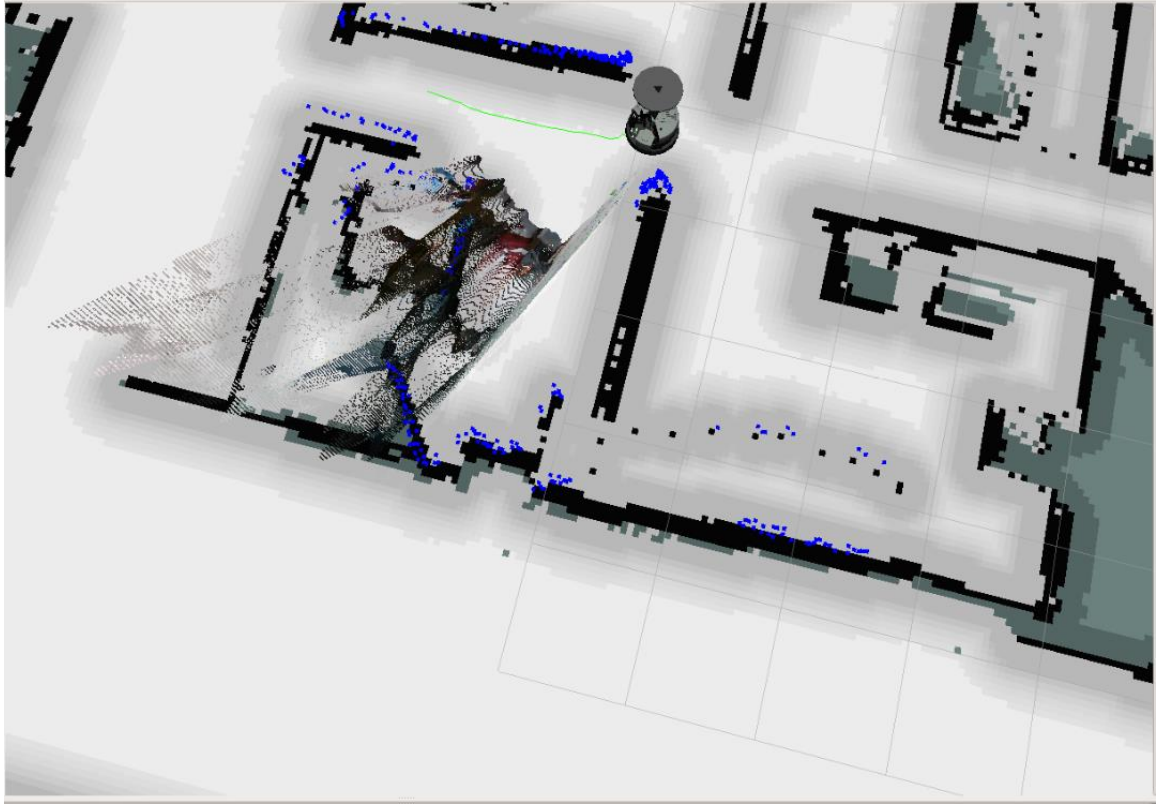
1. Selección de 2D nav goal.

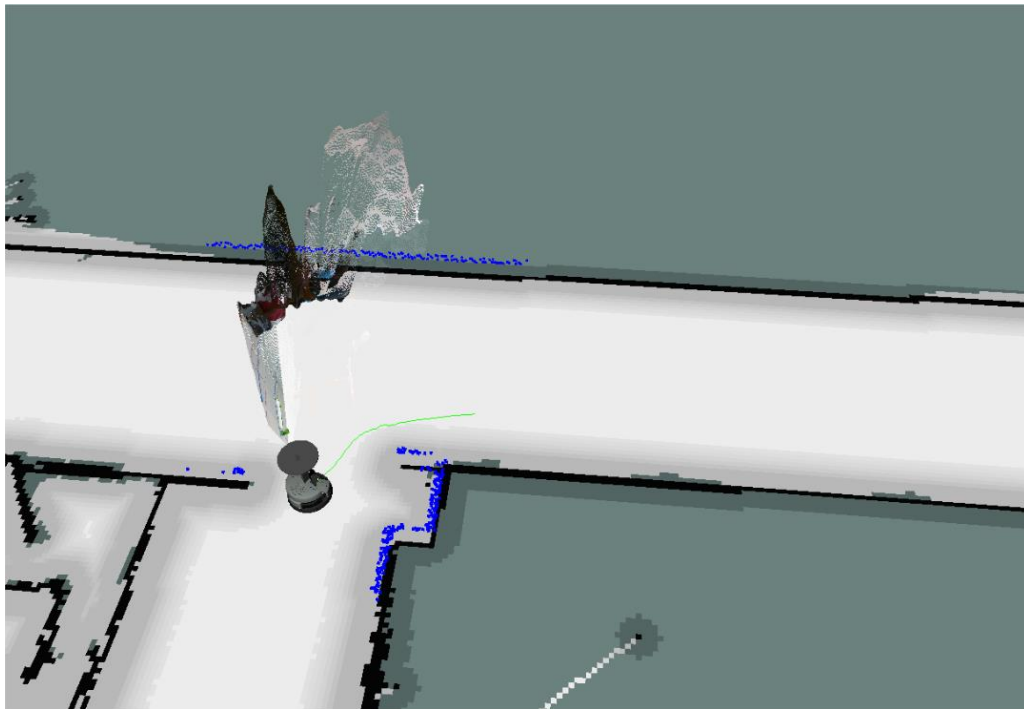
El primer cambio perceptable fue como las paredes y obstáculos del mapa fueron sombreados con una tonalidad gris oscura, lo que se supone es el área navegable adquirió una tonalidad blanca.



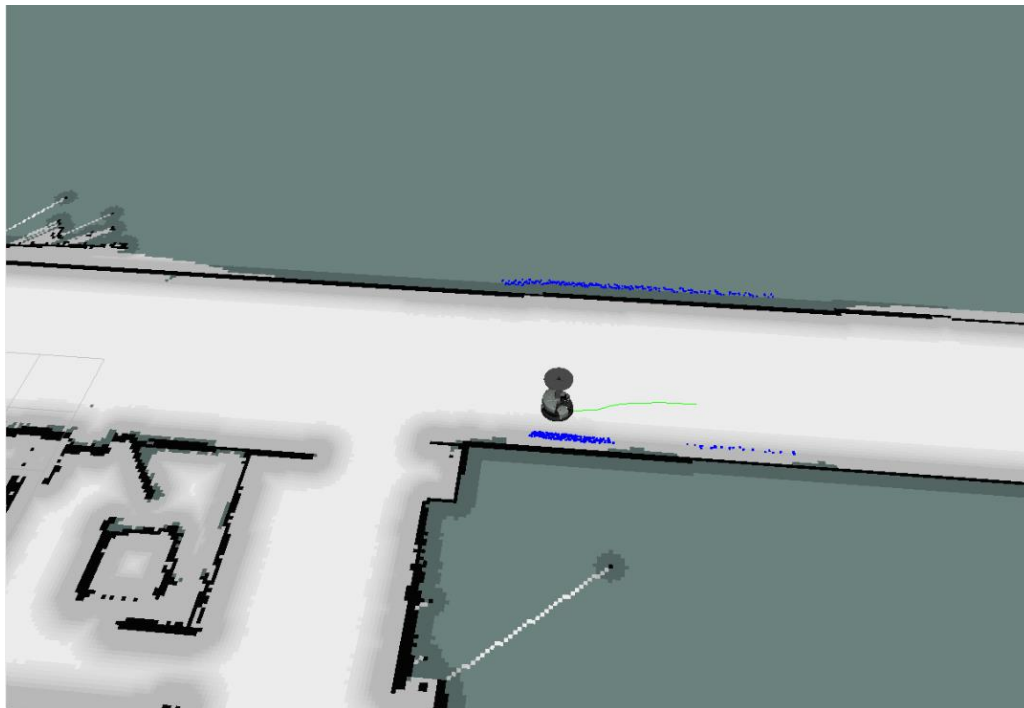
Mientras el robot iba avanzando hacia el punto seleccionado se veía una pequeña línea verde indicando su camino, esta línea algunas veces cambiaba, como si estuviera decidiendo qué ruta tomar, principalmente esto se daba en las vueltas donde la línea se iba acercando o alejando a la pared dependiendo el avance del robot.



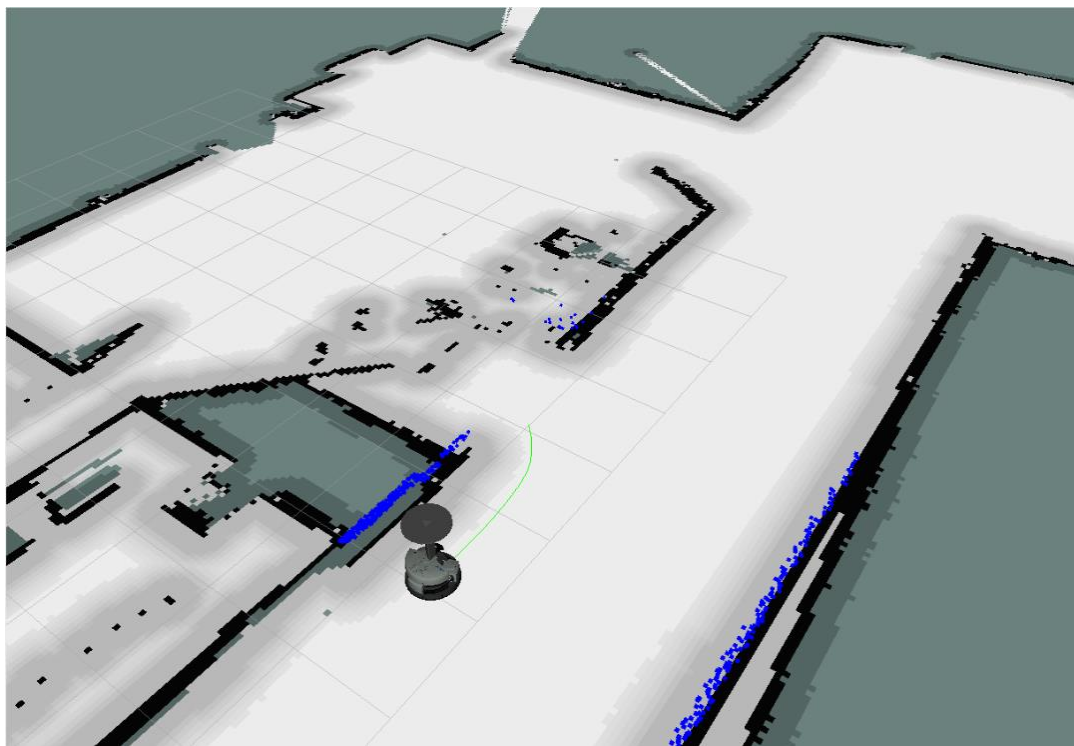
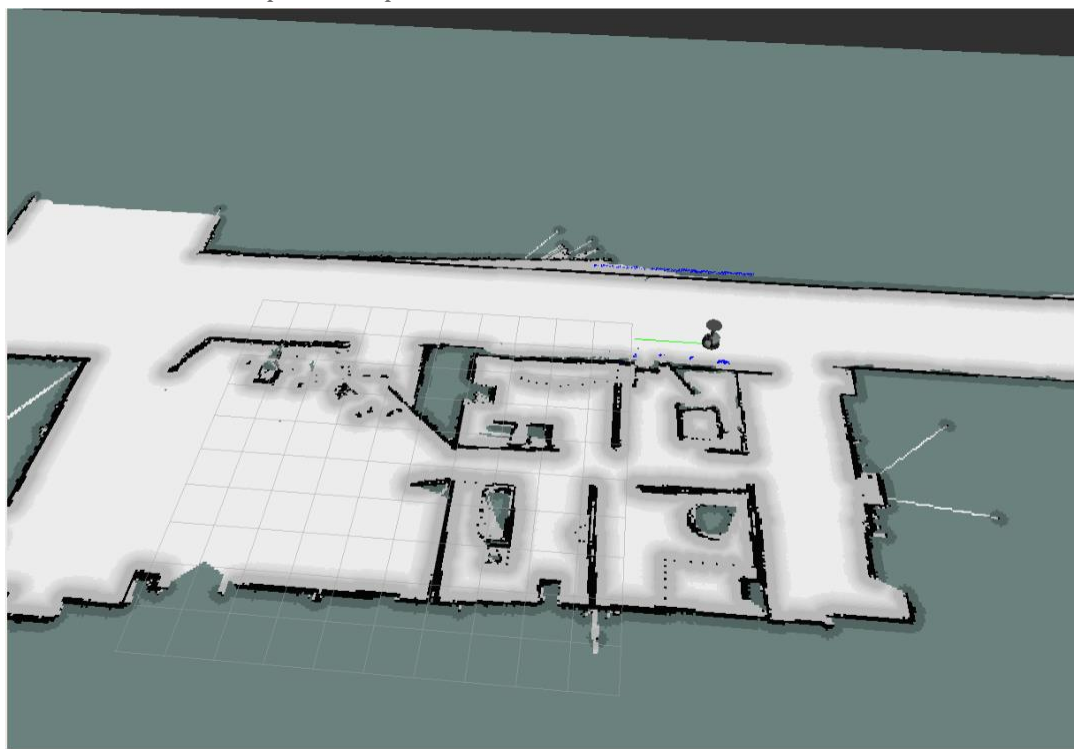




Para quitar la imagen “flotante” que acompañaba al robot se desmarcó la opción de PointCloud2 en la ventana de displays. De esta manera se aprecia mejor la ruta que está siguiendo el robot.



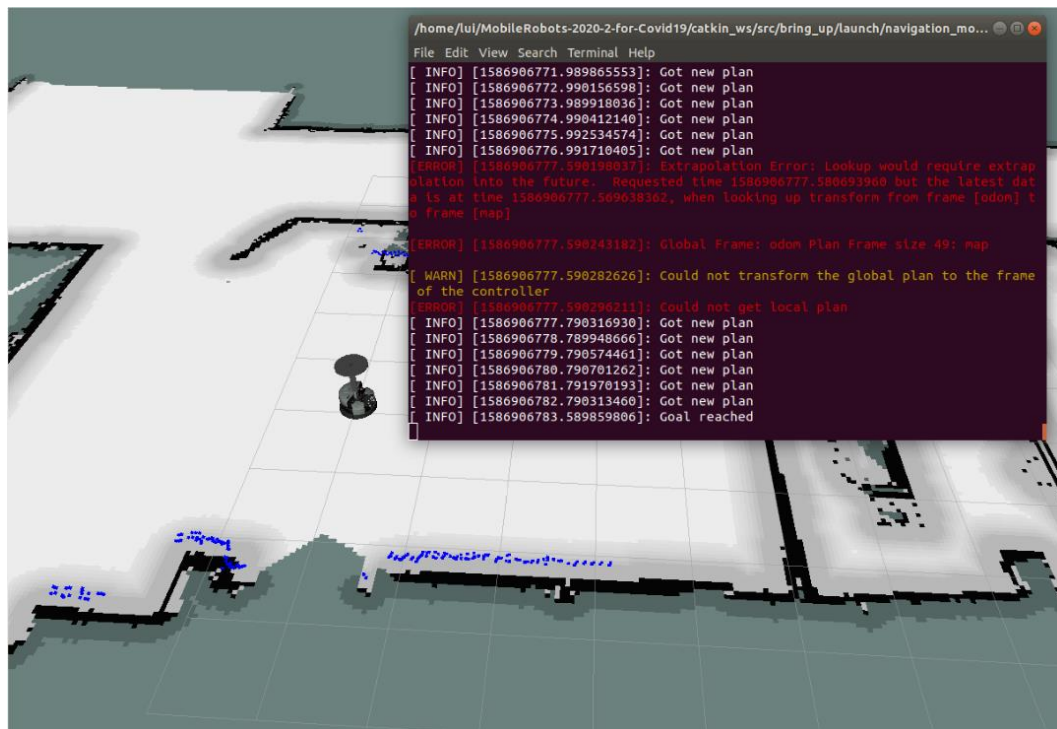
Cuando el robot llega a su objetivo se muestra en terminal un mensaje que dice “Goal reached” es decir, que llegó a la meta propuesta. Se volvió a poner en movimiento hacia un área cercana a su punto de partida.



En la imagen anterior se puede apreciar como el robot pone la ruta para dar una vuelta hacia un callejón sin salida, pero al continuar su camino, corrige la ruta.



Podemos ver como el robot siguió derecho en vez de girar hacia el callejón.



El robot llega a su destino y se informa por terminal, también podemos ver como se genera una nueva ruta cada cierto tiempo, se es de suponer que estas son las correcciones que realiza al momento, como en el caso del callejón.

2. Cambio de archivos yaml.

Al cambiar el archivo costmap lo primero que se puede percibir es el cambio en los bordes del mapa, ahora se tienen unos bordes que se podrían considerar de mayor tamaño y espacios en blancos más estrechos y pequeños. Esto se puede analizar como una reducción en el área transitable por el robot y un aumento en la distancia que debe guardar con respecto a los objetos que se encuentren en el entorno.



Comparando con las imágenes anteriores se aprecia claramente el cambio comentado, aquí se inició otra vuelta para probar el nuevo comportamiento del robot. La ruta parece modificarse menos veces, posiblemente ocasionado por tenerse menor espacio transitable por el robot, al recalcularse menos veces la ruta se hace más eficiente el procesamiento.

La modificación del archivo `dwa_local` incrementa la velocidad del robot, es un cambio bastante perceptible ya que anteriormente se tardaba mucho más tiempo recorriendo el mapa.

OBSERVACIONES

Al principio pensé que la práctica sería más complicada pero gracias al módulo de ROS para la navegación el trabajo parece facilitarse mucho, esto me mete curiosidad sobre los otros módulos que contiene ROS y qué otras cosas pueden realizar.