

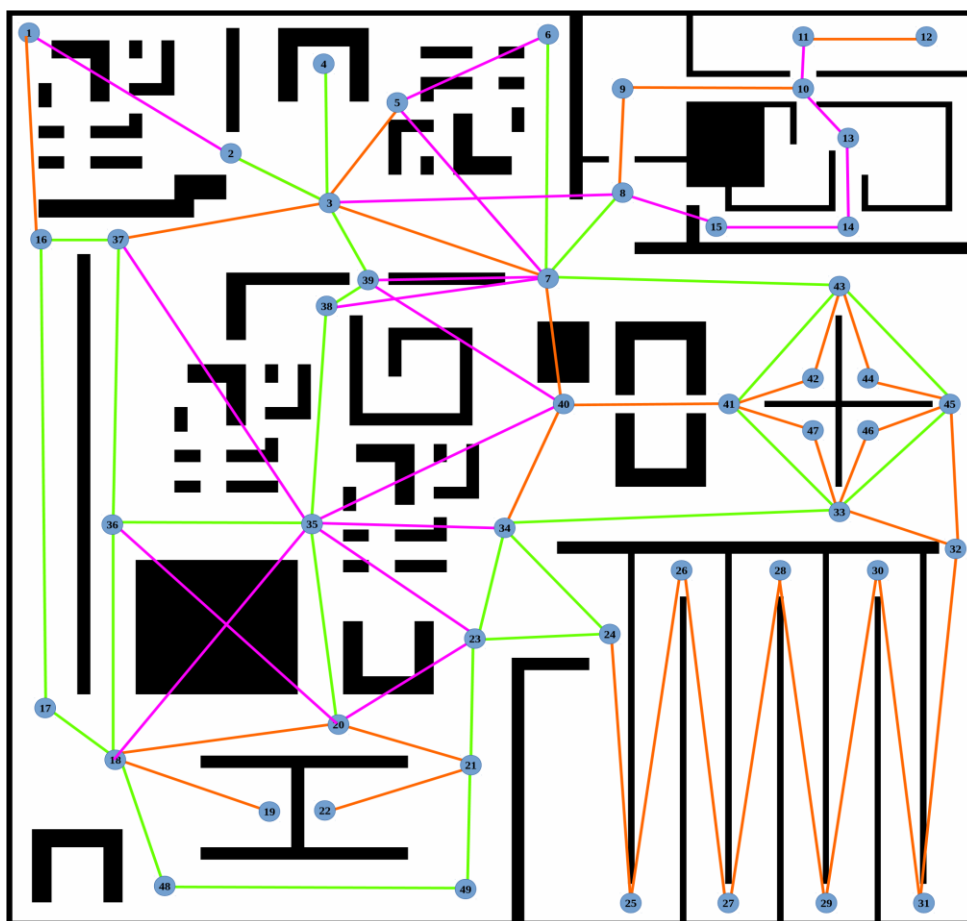
Se desea dotar a un robot móvil de la capacidad de navegación completamente autónoma, es decir, debe ser capaz de resolver problemas como planificar caminos para recorrer largas distancias y, al mismo tiempo, asegurar que realiza toda la trayectoria de forma segura sin colisionar con los obstáculos del entorno. Esta navegación autónoma debe resolver estos problemas en dos niveles funcionales distintos:

1. Navegación global o planificación de caminos: encontrar un camino óptimo entre puntos distantes en el espacio. Este tipo de planificación se realiza a nivel de tarea.
2. Navegación local o reactiva: evitar obstáculos durante el avance del robot en el entorno. Este comportamiento es cercano a la acción (nivel de ejecución) y debe responder rápidamente a los estímulos externos con la información sensorial captada por el robot.

Por tanto, para completar la navegación autónoma es necesario resolver ambos problemas, que ya han sido resueltos en las prácticas anteriores. Para ello, se proporcionan:

- mapa2.pgm: archivo en formato de imagen con el mapa del entorno y la representación de los obstáculos.
- mapa2.m: datos del mapa topológico con la lista de nodos del grafo y la matriz de costes.

El mapa completo del entorno que describen estos archivos que se proporcionan en el Campus Virtual de la asignatura, superpuesto al grafo que representa los nodos topológicos es el siguiente:





Se pide:

1. Implementar un programa de Matlab que utilice la planificación de caminos de Dijkstra (práctica 4) y evite obstáculos con el método de campos potenciales (práctica 3). El programa debe preguntar los nodos de inicio y destino, y representar gráficamente el mapa con la trayectoria del robot. Debe indicar con un mensaje si se ha podido llegar al destino.
2. Realizar la trayectoria partiendo del nodo 1 hacia un destino en la parte derecha superior del mapa. Intentar atravesar la parte inferior derecha del mapa entre los nodos 24 y 32. ¿Es posible alcanzar el objetivo en esas situaciones?, ¿a qué se debe el problema para completar dichos caminos?. Proponer una mejora a la navegación autónoma implementada para resolver esta situación.
3. Si se desea utilizar el algoritmo A* para realizar una planificación de caminos más eficiente desde el punto de vista computacional, es necesario definir una heurística. Se propone definir una heurística consistente y reemplazar el algoritmo de Dijkstra por el A* para la planificación de caminos.

NOTA: si se utiliza la distancia euclídea para determinar la heurística, es necesario calcular una nueva matriz de costes también basada en la distancia euclídea entre nodos, para que la heurística sea consistente. Para construir esta nueva matriz de costes es necesario utilizar la matriz de costes dada como dato del problema, para conocer la adyacencia de los nodos del grafo.