



POLITÉCNICA

MÁSTER UNIVERSITARIO EN INTELIGENCIA ARTIFICIAL

DEPARTAMENTO DE INTELIGENCIA ARTIFICIAL

ROBOTS AUTÓNOMOS

PRÁCTICA 3: REPRESENTACIONES DEL ESPACIO Y PLANIFICACIÓN

ALGORITMO DE PLANIFICACIÓN RRT

JAVIER MORENO VEGA

19 de enero de 2018

Índice

1. Introducción	2
2. Implementación	2
3. Ejecución	2
4. Experimentos	3
5. Bibliografía	5

1. Introducción

La práctica 3 de la asignatura de Robots Autónomos correspondía a Representación del espacio y planificación, por lo tanto este trabajo se centra en unos de los algoritmos de planificación más conocidos el RRT, Rapidly-exploring random tree [2]. Las siguientes partes que componen el trabajo son una explicación de la implementación seguida, una breve explicación de la ejecución del código y por último un par de experimentos realizados.

2. Implementación

Para la implementación del algoritmo RRT se estuvo investigando sobre el pseudocódigo que aparece en diversos artículos de internet pero al final se decidió por utilizar como base el código de Atsushi Sakai [1]. Esta implementación es muy completa y cuenta con su componente gráfico para mostrar los resultados además de diversos parámetros para controlar el algoritmo.

La implementación se ha realizado en python y aunque se usó este código como base se han realizado un par de modificaciones. El código está compuesto por cuatro archivos, `rrt_planning.py` donde se encuentra la clase que contiene al algoritmo RRT con algunos métodos más para mostrar el grafo resultado, `main.py` que es el archivo principal donde se reciben los parámetros y se ejecuta el algoritmo, y un par de archivos de mapas (`map1.py`, `map2.py` y `map3.py`). Los parámetros se controlan en los archivos de mapas donde se pueden definir los obstáculos, el punto de inicio y objetivo; y el valor máximo y mínimo para la aleatoriedad.

3. Ejecución

La ejecución de este algoritmo es muy sencilla y sólo es necesario tener instalada una versión de python 2.7 o superior y los paquetes: numpy, scipy y matplotlib.

Hay que definir el mapa a usar en el archivo `main.py` hacer uso de la sentencia: `python main.py`.

4. Experimentos

Se han realizado un par de experimentos para demostrar la funcionalidad del algoritmo. A continuación se muestran los resultados de los tres mapas, ejecutando el mapa 1 dos veces para mostrar que el camino encontrado puede ser diferente tanto en tiempo como en forma en diferentes ejecuciones.

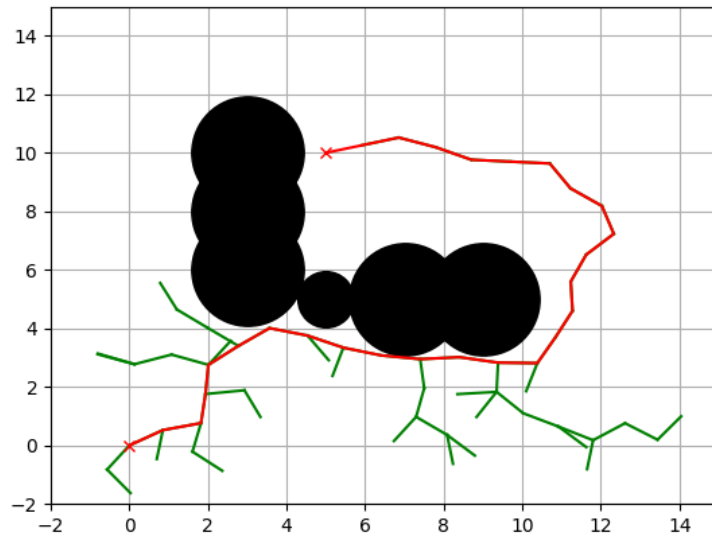


Figura 1: Primera ejecución del algoritmo RRT sobre el Mapa 1.

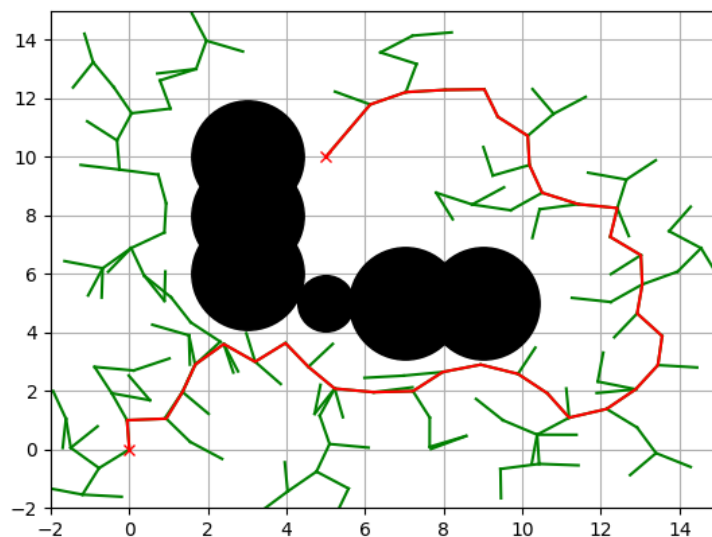


Figura 2: Segunda ejecución del algoritmo RRT sobre el Mapa 1.

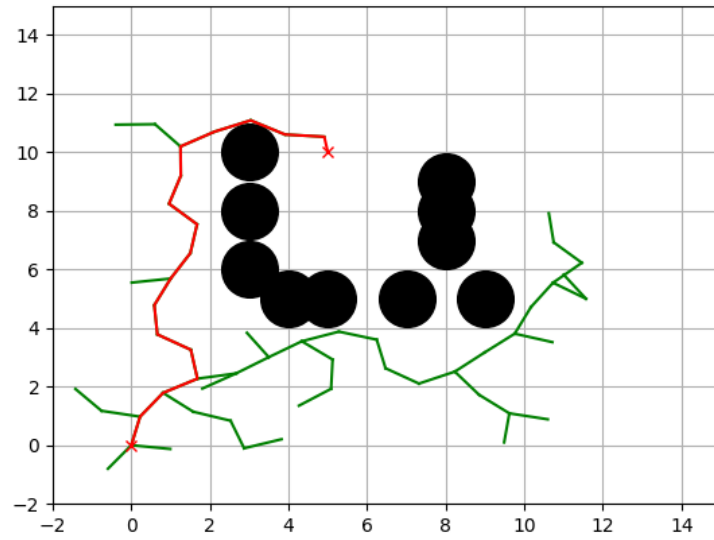


Figura 3: Primera ejecución del algoritmo RRT sobre el Mapa 2.

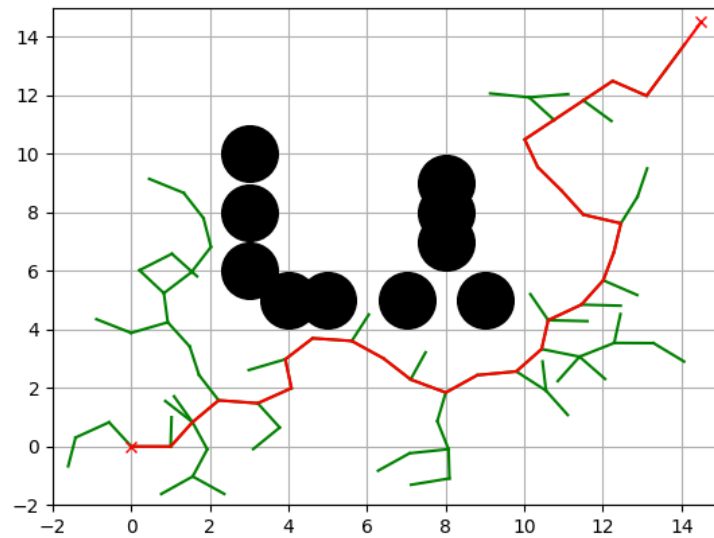


Figura 4: Primera ejecución del algoritmo RRT sobre el Mapa 3.

5. Bibliografía

Referencias

- [1] AtsushiSakai. Pythonrobotics. <https://github.com/AtsushiSakai/PythonRobotics>.
- [2] Wikipedia contributors. Rapidly-exploring random tree — wikipedia, the free encyclopedia, 2017.