



DECSAI

Departamento de Ciencias de la Computación e I.A.

Universidad de Granada



Técnicas de los Sistemas Inteligentes. Curso 2016-17. Práctica 2: Robótica y A*

Guía para poder generar mapas para Stage y visualizarlos en Rviz

Objetivo

La siguiente guía puede usarse como pasos a seguir para generar distintos entornos para simular en Stage y visualizarlos en Rviz. La intención es ser una guía simple para realizar varias experimentaciones con la arquitectura de control implementada en diferentes entornos.

Se explican los siguientes pasos:

1. Generación de la imagen o edición con GIMP (o cualquier otro entorno de edición de imágenes).
2. Preprocesado de la imagen para Stage.
3. Utilización de la imagen en Stage.
 - a. Modificación/Creación del fichero .word.
4. Ejecución de Stage con la nueva imagen de mundo.
5. Pasos para visualizar en Rviz.
 - a. Generar el fichero yaml de la imagen.
 - b. Modificación del fichero yaml para conciliar orígenes de coordenadas de Stage y Rviz
 - c. Visualización del mapa en Rviz.

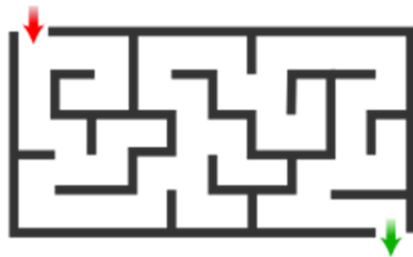


Pasos a seguir

Generación y edición de la imagen con GIMP.

Supongamos que queremos usar una imagen encontrada en Google sobre un laberinto, por ejemplo la de la siguiente url:

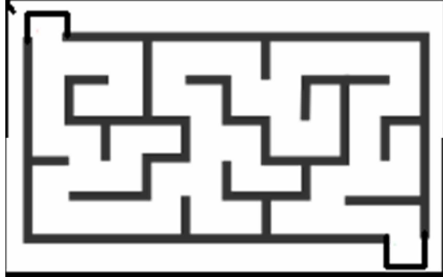
https://upload.wikimedia.org/wikipedia/commons/thumb/8/88/Maze_simple.svg/220px-Maze_simple.svg.png



De 220 x 138 pixels.

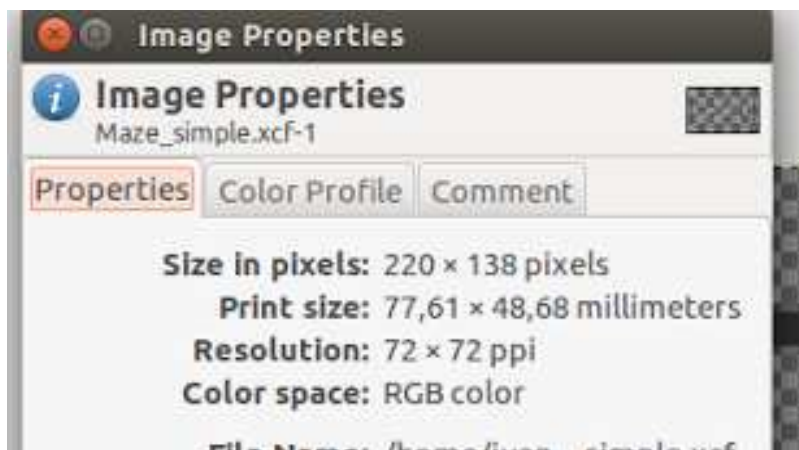
Preprocesado de la imagen para usarla en Stage.

En primer lugar, la editamos con GIMP, la herramienta estándar de edición de imágenes en Linux, le quitamos las flechas roja y verde y cerramos el entorno dibujando un par de cuadrados. Además, rellenamos de blanco todo el espacio que no consideremos obstáculo, para evitar que la transformación de la imagen hecha por Stage introduzca ruido en el mundo. Esto lo hacemos así porque Stage toma el color blanco como espacio libre y el color negro como obstáculo. Finalmente, dibujamos un marco negro que delimite los bordes de la imagen.



Si hemos descargado mistage1516.zip y lo habéis instalado como paquete (como se hizo en la sesión de prácticas), podemos guardar la imagen en mistage1516/configuracion/maps, con el nombre Maze_simple.png.

Las propiedades de la imagen en GIMP nos informan de lo siguiente:



Pasos para usar la imagen como mapa en Stage.

Para usar la imagen en Stage necesitamos crear un nuevo fichero . world. Para ello, copiamos uno ya existente y modificamos los objetos window, floorplan y pr2.

Podemos renombrar el fichero mi-simplerooms.world que está en

mistage1516/configuracion/worlds

a mi-maze.world.



Modificación del objeto window.

La ventana la dejamos con su tamaño original (podemos cambiarla si queremos), no la rotamos y la escala (medida en pixels/metro) la ponemos a 10. Esto significa que tenemos un mundo de 22.0 x 13.8 metros (Size_in_pixels/10)

```
window
(
  size [ 608 300 ]

  rotate [ 0.000 0.000 ]
  scale 10.00 #pixels/meter
)
```

Modificación del objeto floorplan.

El objeto floorplan lo modificamos de la siguiente forma:

```
floorplan
(
  name "mi_maze"
  bitmap "../maps/Maze_simple.png" #ubicacion del mapa
  size [22.000 13.80 1.000] # tamaño en metros
  pose [ 0.000 0.000 0.000 0.000 ]
  #el origen de coordenadas del mundo lo situamos en el centro de
window
)
```

El origen de coordenadas del mundo decidimos situarlo en el centro del mundo, podríamos haberlo situado de manera que el origen de coordenadas del mundo fuera la esquina superior izquierda (como ocurren en GIMP), o la inferior derecha, como ocurre en Rviz.

Para situar correctamente la posición del robot (vamos a situarlo inicialmente en el lugar que ocupaba la flecha roja en la imagen de arriba) estimamos su coordenada inicial a partir de las reglas de GIMP. En GIMP el origen está en la esquina superior izquierda, por tanto el centro de coordenadas del mundo está en la posición (220/2, 138/2). En GIMP la posición de la flecha roja equivale a la (20,15). Esto corresponde a la posición $1/\text{scale} * (-220/2 + 20, 138/2 - 15) = (-9, 5,4)$ en Stage.

Modificación de la posición del robot.

El robot lo situamos en la posición (-9, 5,4 y con una rotación de -90 grados, apuntando hacia abajo).



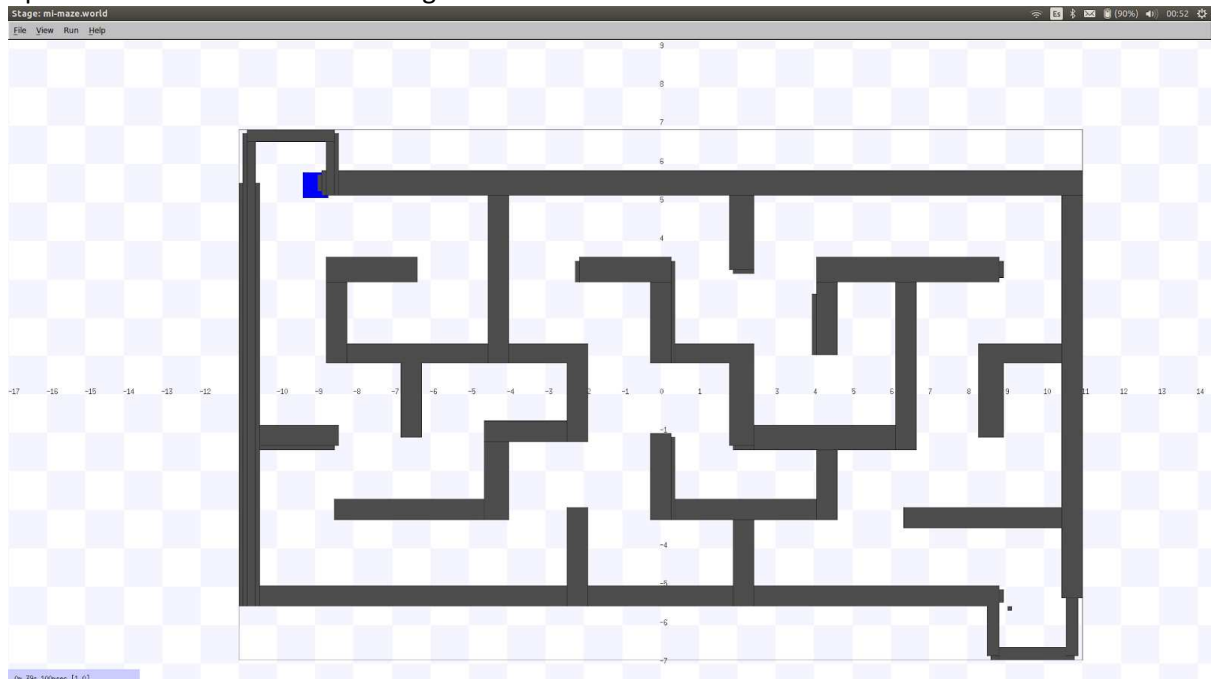
```
pr2( pose [ -9.00  5,4  0.000 -90.00 ] name "pr2" color "blue").
```

Ejecución de Stage

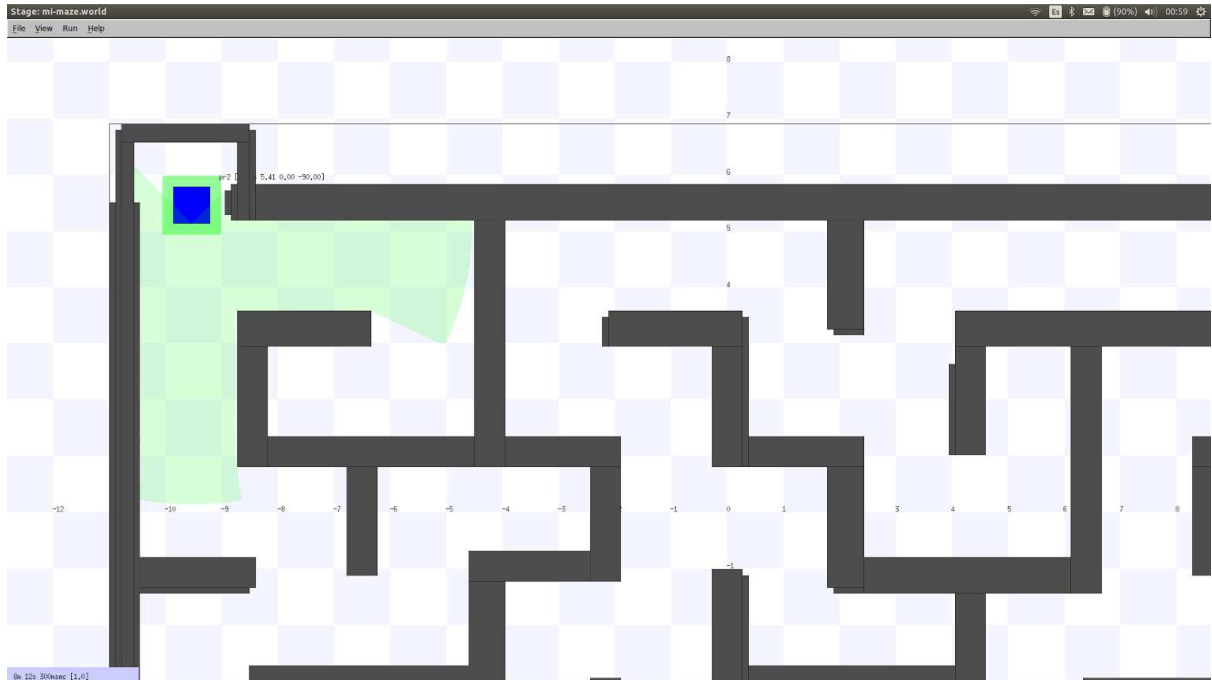
Nos vamos a una terminal y nos situamos en el directorio donde se encuentra nuestro mundo.
Ejecutamos

```
roslaunch stage_ros stage ros mi-maze.world
```

Aparece entonces el mundo en Stage.



Observar que en esa posición el robot está en colisión con un muro, y nos interesa tenerlo más centrado. El robot tiene unas dimensiones de 65x65 cms (ver más arriba en el fichero .world la definición del objeto "define pr2 position"). Basta con desplazar con el ratón el robot a la posición inicial deseada y usamos la opción de Stage File>Save World. Se actualizará el fichero del mundo como vemos en la siguiente imagen.



Observar que la línea pr2(pose) del fichero mi-maze.world ahora ha cambiado y se ha actualizado con la nueva posición del robot.

Pasos para usar la imagen en Rviz

Para visualizar la imagen como un mapa en Rviz necesitamos, por un lado, generar un fichero yaml para poder publicarlo con map_server y visualizarlo en Rviz. Es importante saber que el fichero yaml hay que posprocesarlo para conseguir que el marco de coordenadas de Rviz esté conciliado con el marco de coordenadas de Stage. Seguir los siguientes pasos.

Generar el fichero yaml de la imagen.

1. Ejecutamos roscore en una terminal.
2. En otra terminal nos situamos en el directorio donde se encuentra Maze_simple.png (mistage1516/configuracion/maps).
3. Ejecutamos map_server para que publique la matriz de ocupación de la imagen Maze_simple.png a una resolución de 0.1 metros/pixel (recordar que hemos decidido cuando hemos manejado Stage que la imagen representa un mundo de 22 m x 13.8 m y tiene, 220 x 138 pixels, lo que supone que tenemos 10 pixels por cada metro.

```
roslaunch map_server map_server Maze_simple.png 0.1
```



DECSAI

Departamento de Ciencias de la Computación e I.A.

Universidad de Granada



```
juan@UltraJuanitook: ~/catkin_wsPruebas/src/mistage1516/configuracion/maps
juan@UltraJuanitook: ~/catkin_wsPruebas/src/mistage1516/configuracion/maps$ rosrun
n map_server map_server Maze_simple.png 0.1
[ WARN ] [1458259760.550634505]: Using deprecated map server interface. Please sw
tech to New Interface.
[ INFO ] [1458259780.558876625]: Loading map from image "Maze_simple.png"
[ INFO ] [1458259780.559901889]: Read a 220 X 138 map @ 0.100 m/cell
```

Observar que map_server informa de que ha cargado un mapa de 220 x 138 pixels a una resolución de 0.1 m/cell, o lo que es lo mismo, cada metro cuadrado en el mundo está representado por una celda de 10 x 10 pixels.

4. En otra terminal, vamos a usar map_saver para guardar el mapa publicado en un fichero .yaml. Nos vamos al mismo directorio donde hemos ejecutado map_server y, ejecutamos

```
roslaunch map_server map_saver
```

```
juan@UltraJuanitook: ~/catkin_wsPruebas/src/mistage1516/configuracion/maps
juan@UltraJuanitook: ~/catkin_wsPruebas/src/mistage1516/configuracion/maps$ rosrun
n map_server map_saver
[ INFO ] [1458259872.221117190]: Waiting for the map
[ INFO ] [1458259872.490771832]: Received a 220 X 138 map @ 0.100 m/pix
[ INFO ] [1458259872.490813396]: Writing map occupancy data to map.pgm
[ INFO ] [1458259872.491546332]: Writing map occupancy data to map.yaml
[ INFO ] [1458259872.491647820]: Done
juan@UltraJuanitook: ~/catkin_wsPruebas/src/mistage1516/configuracion/maps$
```

5. Renombramos map.yaml -> Maze_simple.yaml

Modificación del fichero yaml para conciliar orígenes de coordenadas.

Recordar que la línea del fichero yaml

```
origin: [0.000000, 0.000000, 0.000000]
```

se refiere a la coordenada de la esquina inferior izda. del mapa. Para sincronizarlo con el mundo Stage tenemos que decir que la esquina inferior izda. del mapa es (-22/2, - 13.8/2), porque el origen de coordenadas del mundo Stage es el centro del mapa.



Visualización del mapa en Rviz.

Ahora solo nos falta hacer el fichero launch para ver Stage y Rviz con el mapa publicado por map_server.

Si ejecutamos este fichero launch, añadimos el display de footprint y los displays de costmaps, observaremos algo parecido a la siguiente captura de pantalla de Rviz.

Por último, si observamos que los pasillos del mundo son “algo estrechos” para el robot, como se observa en la figura (la inflación en el costmap resulta en un espacio libre “puro” algo angosto para el robot) podemos aumentar la escala del mundo (en Stage) y modificar adecuadamente el fichero yaml como se ha explicado arriba.

