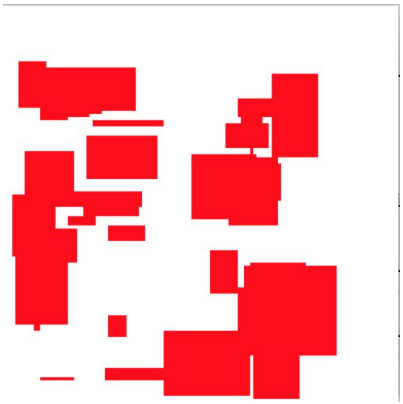
Homework 5: Open-Ended:

1. Monte Carlo Localization on black and white tiles

In this option, you will modify your particle filter code to use a different sensor. You will simulate the "line" sensor on the bottom of the Sparki robot. The line sensor detects whether the surface beneath the robot is black or white. You will need to create a new map class (similar to ObstacleMap) which contains a random grid of black and white tiles. The class should simulate whether the line sensor would return black or white given its position. The particle filter then should use this measurement to weight and update the particles.

We implemented the Monte Carlo Localization on black and white tiles simulation a line sensor on the bottom of the Sparki robot. We located this sensor in the center of the robot.

First of all we created the class TilesMap.py. Which creates a new random map each time you run the code with tiles. Instead of having a grid with black and white tiles we create between 20 and 40 random size rectangles placed inside the map. Using this way we show the expected result too. This would be an example of a map: (We changed the colour of the tiles to red so we don't confuse them with the particles)



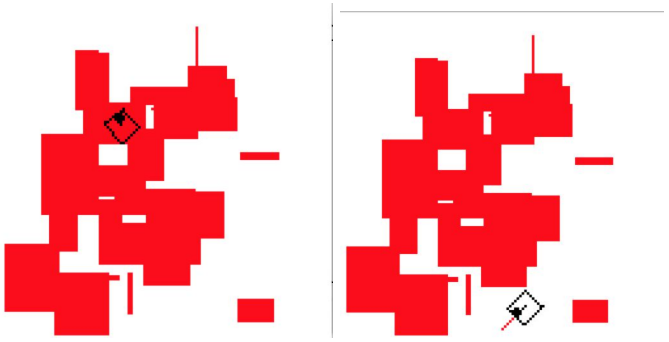
Also, in TilesMap.py class we created a function get_tile(self, cx, cy) which return if a specific position is a tile or not. A value of 200 if it is not a tile and a value of 900 if it is a tile.

Then we modified the ParticleFilter.py module in order to update the particles weight using the new line sensor reading.

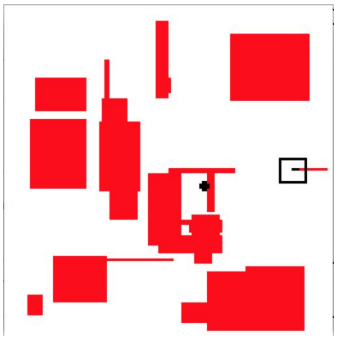
Finally, we modified the old localization.py module into the localization_tiles.py module to run our code. We adapted the new TilesMap.py into this module.

Results

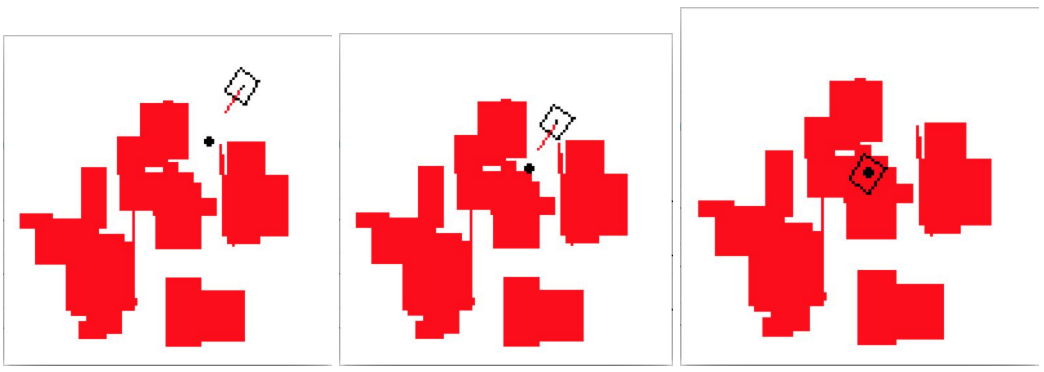
We can see in the following images that the particles converge good to the robot's position. The robot has gone through some tiles and no tiles in order to converge better.



We also see that sometimes it doesn't converge good. In this case, the particles got stuck in there and they would no go outside of that white rectangle until the robot goes through some tile.



In this last case the particles did not converge good. One way to make them converge good is make the robot follow the particles once they get stuck in a tile border. After that, the particles are in a good localization.



Another thing to say is that if the particles converge good and we start moving in a tile or outside the tile, the particles start to diverge until the robot sensor reading change. We can see that behavior in the following images.