PROBLEM ANALYSIS

The Exploration problem is bigger than Navigation.

Furthermore we don’t known nothing and we have to move blindly.

For this phase we can use the Console and Robot as described in the Navigation phase, so I haven’t to repeat the Model analysis, at least for the entities structure.

HOW TO EXPLORE

The first exploration idea is to use the left-hand wall follower algorithm.

This algorithm tell that, if an unknown maze is simple defined, if a human maintain the left hand on the wall, it will, sooner or later, find the exit of the maze.

A simple defined maze is a maze where all the wall touch the external bounds of the maze. So we haven’t loops.

But if I think to a house I always have objects surrounded by walkable areas. So in the real world I haven’t simple defined environment, and this algorithm isn’t applicable.

So this is the Idea:

* I work on discretize map, like in the Navigation problem, cells are square-shaped and uniform in size;
* Robot fits perfectly the single cell;
* A cell can be : Object, Clear or None (aka not explored);
* At runtime I store both Object and Clear cells, when the map is totally explored I save only Objects;

At the beginning all cells are marked as None (not saved) except of the start cell that is obviously Clear.

The Robot explore until:

* There are no more None areas;
* The user stop it;

Everything is based on 3 method:

* First of all we have to findLeftWall, this is the first method. When I’m lost in a clear area I have to find a wall to follow; It just move forward until hits a wall.
* When I found a wall I follow it. So Explore do exactly this, until it’s possible, the Robot move forward. When it find a wall it turn right. When find an already visited cell start a little search to find the nearest not explored area;
* Then moveAway and navigate (with the best path including only explored cells) to the unexplored area.

At every movement the robot stores a Clear Cell, every time it sense (with the sonar) an object it stores an Object Cell.

Every Cell must be stored while the exploration is running to compute the moveAway path, but to avoid the transmission of a very big map, Robot sends data every time it store a Cell.

This “data” must be a message to avoid possible lost, and can be in the form of exploredata : exploredata ( position ( X , Y ) ).

Obviously Robot don’t know the maximum area of the map, so during the exploration this parameter is updated with the max and min coordinates for each axes.

At the begin the start coordinates of robot are assumed as 0,0. At the end, each point is shift to position the origin in the top left corner of the map.

In the initial tests can be useful to set start position and maximum boundaries in order to show the exploration progression.