Homework 5: Car Tracking

Part I. Implementation (15%):

* Part1:

First, we get the size of the map by calling “self.belief.getNumRows()” and “self.belief.getNumCols()”.

For each position in the map we convert it into x,y location

And calculate the distance to our agent,

Have this distance, with observedDist and const.SONAR\_STD we can use “util.pdf()” to get the “probability density of a Gaussian distribution”

Therefore we can use pervious belief P(Ht|E1:t-1)\* P(Ht|Et) to get new belief

P(Ht|E1:t). and we can use “self.belief.setProb()” function to update it

The last we normalized the probability to make it sum equal 1.

一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

自動產生的描述

* Part2:

First, we get the size of the map by calling “self.belief.getNumRows()” and “self.belief.getNumCols()”.

And we use these two information to create a new belief.

\* the form of self ((oldx,oldy) , (newx,newy)) : prob

By the transProb we can get the probability of transfer old position to new position.

Furthermore we can use “self.belief.getProb()” to get old probability P(Ht|E1:t) and multiply it with transfer probability P(Ht+1|Et) to get newbelief P(Ht+1|E1:t) and (use.addProb()) function to get add it to new belief.

The last we normalized the probability to make it sum equal 1.

一張含有 文字, 螢幕擷取畫面, 軟體 的圖片

自動產生的描述

* Part3-1:

First：

For each particle in now self.particles(), we first convert it position into x,y and calculate the distance to our agent.

Have this distance, with observedDist and const.SONAR\_STD we can use “util.pdf()” to get the “probability density of a Gaussian distribution”

With the probability of old particles, we can calculate the new particle position’s probability and record it into the particle\_dict.

Second：

For each new particle we get before we use “util.weightedRondomchoice()” to get the new position of each particles and record it into the dictionary.

Last we update self.particles with the new particle we get.

一張含有 文字, 螢幕擷取畫面, 軟體, 作業系統 的圖片

自動產生的描述

* Part3-2:

We first initialize an dictionary to record the new particle distribution. And we use the function “util.weightedRondomchoice()” to get the new position of each particles and record it into the dictionary.

Last we update self.particles with the new particle we get.

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

Part II. Question answering (5%):

The problem I encountered is the math in this assignment, when I see the math in the spec I have no idea about it. To solve this problem I read the more information in the spec and search on google to realize the math and why calculate in this way. When I start to write the code I remember the what goal to do in each part, but don’t know how to start it. Therefore, I start to read util.py and the functions in it and know the structure of the class and know how to use this function to help me finish this assignment.