Project 2 CSC C85

Adam Wu (1004266741)
Joseph Boychuk (1005071468)

Handout Answers

1. Assuming m and n counts the number of intersections (like in the starter code), then 4mn probabilities in total. If m and n refer to the number of grid cells, then it would be 4 * (m - 1) * (n - 1).

2. 1 / (4mn) which is 1/60 in our scenario.

3. For our colour sensor, we compare the sensor reading to a reference value. To determine the reference value of each map colour, we added a calibration routine. For each map colour, we take multiple colour sensor readings and at different places on the map, and then use their average as the reference value. To determine the colour, we add up the absolute difference between the r, g, and b components and choose the reference value that had the smallest total difference smaller than a threshold (tolerance).
This should work since the readings at different spots of the map for the same colour are a little different and by taking the average of multiple values and different locations, it is not overly sensitive to 1 particular spot on the map.

4. Our probability function works in 2 parts that we call SENSE and MOTION. During the SENSE step, the robot compares the scanned colours at an intersection to the values in every intersection in the map array at each orientation. If the colours match, the robot raises the probability at that intersection and orientation by a factor based on our sensor reliability (measured through extensive testing). If the colours do not match, the probability of that intersection at that orientation is reduced by the opposite factor (chance the sensor was wrong). This preserves small values and allows us to adjust the probability by our confidence in the sensor's accuracy. We found that the colour sensor after denoising was about 95% (approx 1/20 incorrect) accurate, so matching spots are multiplied by 95 and non matching spots are multiplied by 5 to reduce error from very small numbers.
During the MOTION step of the function, the movement is evaluated. The expected movements are a move forward or a 90 degree turn. In both cases, we tested the robot separately in its ability to perform these actions. We found that in these cases, there were other possible outcomes such as accidentally turning too

much and ending up at other intersections or in the wrong orientation. To compensate for these, our motion model has probabilities for each tested outcome. The robot performs a motion and then informs the probability function of the motion it just performed (move forward, turn). The beliefs are then adjusted based on the success report of that motion according to the motion model. The beliefs are then normalized for the next iteration of the localization process.

5. Our Robot's exploration strategy is to find the corners of the map. We know that localization relies on the use of landmarks. In this case, the important landmarks are the buildings, the road, the intersections, and the boundaries. Since the important landmarks for gathering information about position are the buildings and the boundaries, we decided the best place to gather the most information is at the location where these landmarks have the highest concentration: the corners. The corners provide the most information out of any location because they contain the same amount of road, intersection, and building information but also have 2 boundaries for more positional info. There are only 4 corners on the map, so it narrows down the potential intersections to just 4, allowing much quicker convergence in the localization process.

6. The robot cannot drive past a boundary so when it encounters it, our robot drives backward to go back to the intersection it was on a moment ago. So that means the robot should be facing the boundary at an intersection next to a boundary. Therefore we increase the belief for the orientation that faces a boundary at those intersections. This does assume that our robot encountered the boundary by driving along the road properly from the intersection, as that informs the orientation of the beliefs to change (only orientations that face boundaries).

7. The beliefs are rotated about the intersection but still follow a defined motion model in the same way that updates from driving forward rely on the motion model. Since we tested turning, we learned that turns are about 99% accurate, and as such we update the beliefs using this factor that the turn was successful and the rest of the factor is distributed to the other motion case which is that the turn failed and the robot is still in place. This means every belief value is the belief it rotated from multiplied by 99 and the small remaining percentage.

8. After normalization, the robot has a set of beliefs for every intersection and orientation. If a belief is greater or equal to 95% we consider the robot localized. We found that the robot reaches converging values around 96%. Setting the threshold higher makes the algorithm continue to run in an attempt to reach those high values, and in some cases prevent it from ending if the beliefs never

get increased that high. The uncertainty in the motion and sensor accuracy make it hard to reach extremely high values, so this is an acceptable balance between reasonable certainty and efficient localization.

9. We found the sweet spot for the colour sensor was very close to the surface it’s scanning, so we placed the colour sensor very close to the ground.
   We made the distance between our wheels wide enough so it isn’t sensitive to slight differences in the direction the wheels are facing. Thus driving forward is more reliable. It also makes our robot more rigid since if they are too close, the robot could sway to the side a bit. We added 2 bars between the wheels so they are more securely connected to the EV3 brick.
   We added a track ball at the front of the robot so it rotates more smoothly and it helps keep the robot stable.
10. The accuracy of the colour sensor is the most important parameter for the robot because it is the hardest to adjust in a positive way and open to the most influence from external factors. Even if the robot positions itself perfectly in every way, if the colour sensor fails or reads the wrong colour it undoes a lot of the progress the robot has made towards localization.

11. External factors can definitely affect the ability to find itself.
    Most importantly, the robot relies on the colour sensor to identify landmarks on the map but the sensor is sensitive to different lighting conditions. The lighting condition will change the values it reads. Without calibration, it’s only determining the colour based on some fixed expectation of a colour rather than what that colour actually looks like in that environment. This is why we added a calibration step so we compare against a more accurate expectation of the map colours in the robot’s environment.

12. On average, our robot takes about 7 intersections to converge to a localized position. It takes between 1 - 2 intersections to find a good orientation from the starting position, then has to travel around 5 - 6 intersections from its position to the corners where it usually immediately converges well.

13. Since the exploration method for our robot is to find the corners of the map, starting in a corner oriented away from it usually takes the longest as the robot usually has to travel the furthest distance to get to another corner. Although this is technically one of the hardest start positions, it does not vary much from the difficulty of other spots on the map.

Feedback

1. Yes - Super satisfying to watch it fail over and over and then suddenly it all just works, especially considering it does not have to be perfect all the time. Watching it slowly converge is fun!

2. Yes - Very insightful working with the light sensor and seeing how fickle it is. The gyroscope sensor was also annoying and both made us appreciate error handling and properly informing users.

3. Yes - Although the motors were not as uneven as we expected, it was a challenge having to deal with battery life especially during early stages of testing.

4. Yes - Made so many mistakes around the computation step, finally getting it to work was that much more satisfying.

5. Somewhat - Since the management of the uncertainty was built into the localization algorithm a little bit, it was more about finding a convincing motion model.

6. Yes - Hopefully this robot has higher quality parts that don’t break with no indication, and if they do it is less frequent!

7. This was a very fun assignment but definitely challenging to complete alongside other assignments and courses. The MATLAB assignments suffered because of working on the robot. Your warning at the beginning of the semester was no joke, this course is very time consuming!