Navigation

**Description**

The test mainly checks whether the navigation system can function desirably in the final project. Our current Navigation System can localize the robot to waypoints with low errors. However, the large time consumption of it will lower the performance of the robot. The Navigation System software will be modified and different versions of it will be tested and evaluated.

For the test, similar to lab 5, we will pass some random way points to our design. We expect the robot to move to these way points in order. We expect to observe the error is within the same range of the current design. But we want the robot to complete the map for a much shorter time. Moreover, since the final project asks the robot to run the laps as many as possible, we will also conduct a test on whether the navigation system can assist the robot to run a map for multiple times and finally end with small errors.

**Test1**

**Date :** 2021/3/22

**Tester:** Shichang Zhang

**Author:** Shichang Zhang

**Hardware version:** 1.3 (in Part 2.5 of [Hardware Document](https://docs.google.com/document/d/11jkA_S_xBqyCbcn2NyMuM-OMDEybDfRy/edit#))

**Software version:** 1.3 (in Part 7.0 of [Software Document](https://docs.google.com/document/d/19JaY5629aUu4Y4rjoQJ-jWyeQLqNSAcr/edit))

**Test Purpose:**

Determine whether the navigation system can help the robot to go through the waypoints in order and finally end at (1,4) with a small error. .

**Test Procedure:**

We will place the robot at (0.5, 4.5) as the start point. And we will pass a series of waypoints to the robot. We expect the robot can go through these waypoints in order and finally stop at (1,4) with small error. We also expect the time cost of the navigation process to be low. .

1. The robot is placed at coordinate (0.5, 4.5), that is (0.1524, 1.3716) in meters.
2. The robot is set to orient to 180 degrees.
3. Pass the input waypoints to the robot.
4. Set the forward speed to 500, rotate speed to 200, localize speed to 200.
5. Start the odometer. Initializing the odometer with value (0.1524, 1.3716,180).
6. Run the navigation system, start the timer.
7. Record whether the robot goes through all points in order.
8. When the robot ends at (1,4), stop the program, end the timer.
9. Print the final translation and angle indicated by Webot.
10. Record the time taken for the navigation.

**Test Data:**

|  |  |
| --- | --- |
| Trial# | way points (ft,ft) |
| 1 | (1,4), (4,2),(6,5),(4,5),(11,1),(14,4),(8,4),(1,4)  Figure 1.1 The navigation path on the island of trial 1 |
| 2 | (1,4), (5,2),(9,4),(7,5),(9,1),(14,4),(11,5),(14,2),(4,5),(1,4)  Figure 1.2 The navigation path on the island of trial 2 |
| 3 | (1,4), (4,5),(9,1),(6,5),(14,2),(14,4),(8,4),(4,1),(1,4)  Figure 1.3 The navigation path on the island of trial 3 |
| 4 | (1,4),(3,1),(11,1),(11,5),(8,4),(10,2),(14,2),(7,5),(5,2),(1,4)  Figure 1.4 The navigation path on the island of trial 4 |
| 5 | (1,4),(6,2),(8,5),(9,1),(11,5),(13,1),(14,4),(10,1),(6,5),(3,2),(1,4)  Figure 1.5 The navigation path on the island of trial 5 |

**Expected Result:**

|  |  |  |  |
| --- | --- | --- | --- |
| Trial# | Translation (m,m) | Angle (deg) | Timer (HH:MM:SS) |
| 1 | (0.3048,1.2192) | 270 | 00:05:00>time |
| 2 | (0.3048,1.2192) | 270 | 00:05:00>time |
| 3 | (0.3048,1.2192) | 270 | 00:05:00>time |
| 4 | (0.3048,1.2192) | 270 | 00:05:00>time |
| 5 | (0.3048,1.2192) | 270 | 00:05:00>time |

**Test Results:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Trial# | Translation (m,m) | Angle (deg) | Timer (HH:MM:SS) | Translation error(m,m) | Angle error (deg) |
| 1 | (0.3059,1.2193) | 272.1 | 00:03:24 | (0.0011,0.0001) | 2.1 |
| 2 | (0.3070,1.2198) | 270.3 | 00:03:50 | (0.0022,0.0006) | 0.3 |
| 3 | (0.3035,1.2165) | 270.8 | 00:03:11 | (-0.0013,-0.0027) | 0.8 |
| 4 | (0.3089,1.1288) | 270.5 | 00:03:34 | (0.0041,-0.0004) | 0.5 |
| 5 | (0.3078,1.1289) | 270.9 | 00:04:08 | (0.0030,-0.0003) | 0.9 |

**Test Report:**

The test is performed 5 times for different input waypoints. The pass rate is 100%. We expect the robot to travel to each waypoint in order and end at (1,4) with limited translation error and angle error. In trial 1,2,3,4, we design the paths that have several waypoints which the distance between these two consecutive waypoints are very large. We observed some deviation during long distance travelling when performing these trials, and we also observed that the robot is able to reduce the deviation considerably by localizing before arriving at the next waypoint. In trial 5, we design a path that has many waypoints and requires the robot to turn a large angle when it is trying to be navigated to the next waypoint. From the tested output, we can see that the navigation system handles these paths and generates little translation and angle error at the final waypoint. We also observe that the time taken for each trial is around 3-4 minutes, which means the robot can accurately travel through 9-10 waypoints quickly given there is no obstacle on the path. Overall, the navigation system gives really little error and it also runs fast. .

**Conclusion:** Pass

**Action:** None

**Distribution:** software development