Robotics – Assignment 1 – Dock the Roomba

David Wilkie
September 9, 2008

1 Problem

When Roomba has completed its cleaning cycle or when the battery is running low, it automatically returns to its Home Base (figure 1) to recharge. The goal of this assignment is to write a program for the Create robot to find and dock with its charging station without human intervention. You may assume that the Home Base is located directly next to a wall, its immediate vicinity is unobstructed by obstacles, and all walls within the environment are connected, with no gaps. You should not assume that the Home Base will always be in range of the infrared sensor on Create, so you will need to decide on a strategy to guide the robot towards its goal without that information until a signal is detected.

from http://wwwx.cs.unc.edu/

2 Method

The difficulty in docking the roomba is in orienting it toward the base station given its knowledge of the sensor field. My strategy is to first find a wall and follow it until the base station is found. If the robot always performs right-wall following, then we know the robot’s orientation relative to the base station when it is found. The robot will move away from the IR field until it has found a wall and can orient itself. This is done by the robot by moving straight when no bump or IR are detected and turning at a sufficient radius when it detects an IR field (to leave the IR field). This can be seen in Video 2.

Wall following is done using the bump sensor and the wall detector. It is calibrated for low reflective hallways (such as those in Sitterson). Initially, the robot is bumping against the wall: in this case, and whenever it bumps into a wall, it turns counterclockwise in place until it gets a wall sensor reading. Whenever the robot detects a wall with a signal greater than 20 (determined to be the value it gets when close to SN’s walls), it moves straight. If it does not detect a wall, it turns right at a small radius until it detects a wall or bumps. This behavior can be seen in all videos.

Once the basestation has been found by receiving the force field IR signal, the robot moves around the perimeter of the force field. This is achieved by behaviors in two cases: 1) if the robot is in the force field, it moves along a curve of sufficient radius in the right direction (moving out of the forcefield), and 2) if the robot is outside the forcefield, it moves along a curve in the left direction of sufficient radius (which takes the robot back into the forcefield). This behavior traces the outside of the forcefield and continues until the Red and Green forcefield is found. This behavior can be seen in all videos.

Once the Red and Green field has been found, two methods were developed to dock.
2.1 Method 1

First, I used the method of following the right side (bordering the Red field) of the Red and Green forcefield. This was done the same way as following the force field. This method resulted in the robot being slightly misaligned when it reached the base station, however. A behavior to dock once in this misaligned state was designed: the robot reverses along a radius for a short period of time, then goes straight along a slightly larger radius for approximately the same period of time. This behavior makes the robot translate in the direction of the center of the basestation while testing all points along the line for docking ability. This behavior can be seen in Video 4.

2.2 Method 2

The second method is based on how the Roomba demo docks. It uses the same procedure to find the Red and Green field. Once the field is found, it follows two simple rules: 1) if the robot detects the Red and Green field, it moves straight, and 2) if it leaves the Red and Green field, it turns right until it finds the Red and Green field again. The code is designed so that the robot knows which side of the Red and Green field it entered from, but this is actually not needed as the robot always approaches the base station from the same side due to the wall following. This method aligns much better and faster than the first method. This can be seen in Videos 1-3.

3 Results

Videos of results can be found at http://www.cs.unc.edu/wilkie/assignment1.html.

4 Implementation

I based my project on the Drive demo for the Command Module. The Command Module was used for controlling the robot, and an LCD was used for debugging. WinAVR and lcd4 were libraries used. The makefile assumes the usb cable is in Com9.

5 Source Code

Source for the project can be found at http://www.cs.unc.edu/wilkie/assignment1.html.