Authors: Michael Bonnet, Allison Gardiner, Noah Walker
Class: CSE 4360-001 FA2021
Date: 11/30/2021
Assignment: Project 2

Project 3

I. Background/Introduction

When our team received the specification document for Project 3 including several suggested projects, we did not find any of them to be quite to our liking, and so began brainstorming what we'd propose instead.

Eventually, we settled on proposing that we design and implement Braitenberg vehicles, simple robots that directly map sensor readings to actuators in ways that create different behaviors based on whether the robots accelerate or decelerate towards signals, and turn towards or away from signals. Combinations of these two criterions' two variations result in four behaviors:

1) "Fear", where the robot will turn away from signal and accelerate the closer it is to the signal, aiming to come to a stop when it has turned fully away from the signal and no longer detects it, as if it "fears" the signal.

2) "Aggression", where the robot will turn toward the signal and accelerate the closer it is to the signal, as if it were trying to "attack" the signal.

3) "Love", where the robot will turn toward the signal and decelerate the closer it is to the signal, as if it "loved" the signal and wished to be peacefully near to it and face it.

4) "Explore", where the robot will turn away from the signal and decelerate the closer it is to the signal, as if it were "exploring" the world and wished to observe signals then seek new ones.

Our proposal was approved, and we got to work.

II. Robot Design

Similar to our designs for Project 1 and 2, we followed the Medium Motor Driving Base assembly instructions found on the ev3 website. We chose to do this for similar reasons to Project 1:

- Copying a known configuration reduces time spent on robot design and construction

- Copying a manufacturer-recommended configuration potentially reduces time spent adjusting robot design
- Two wheels simplifies kinematics of the robot
- Three points of contact with the ground nigh-guarantee stability

Beyond the above design and reasoning, we of course had to add sensors.

The issue facing us at the start was that none of the sensors available to us detected both range AND direction. This necessitated splitting the detection responsibility between two sensors, and integrating their readings to inform robot movement.

For range detection, we used an **Ultrasonic Sensor** pointed forward from the robot, mounted low to the ground. We chose this forward position so that the robot would simulate Braitenberg behaviors as closely as possible, and we chose the low position so that it would not miss any narrow-bodied goals - and because it was the easiest mounting point available. Using the ultrasonic sensor for distance detection meant that we would be able to detect most any low-lying signal sources, such as other Braitenberg vehicles.

For direction detection, we used an **Infrared Seeker** similarly pointed forward from the robot, but mounted higher up, roughly in line with the top center of our EV3 brick. We chose the forward position again so that the robot would simulate Braitenberg behaviors as closely as possible. We chose the high mounting position so that the seeker would be more level with where we intended the infrared signals to come from - mounted on top of the Braitenburg vehicles themselves.

We made three identical versions of this design on three different EV3s so that we could attempt to observe interactions between Braitenberg behaviors.

III. Signal Source Design

A unique factor in our project was the necessity of emitting signals for the Braitenberg vehicles to react to. Given that we had to use two different sensors that sense two very different things, we needed to emit those two different things from the same point.

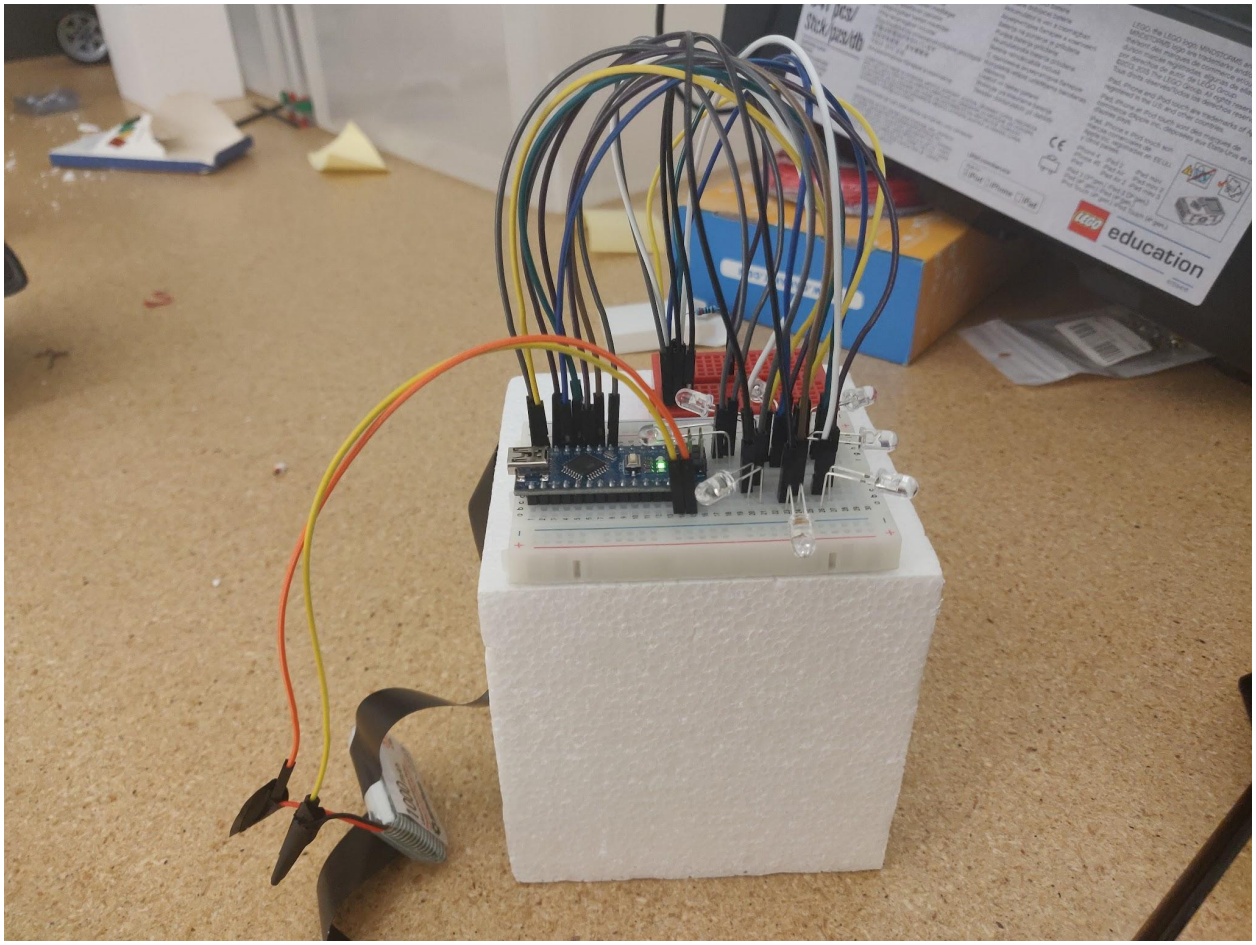
For our ultrasonic sensor, things were relatively easy. Any fist-sized or larger object at the same level as our sensor reflected sound back at the sensor during preliminary testing, and our sensor was near to the ground - gravity is on our side, so no need to suspend anything. So long as we had a decent-sized object, we'd be fine. Therefore, whatever that object *is* would be informed by how we address the other signal.

For our infrared seeker, things were significantly more difficult. To begin with, the ev3dev and MicroPython/pybricks environments that we had grown used to in our previous projects did not directly support the NXT Infrared Seeker V1, only the V2, which we did not have access to. Finding a solution to this challenge took several days, and so our infrared signal generation

method development had to essentially “fly blind” in parallel with fixing and/or integrating the sensor. This hampered overall effectiveness of the end solution.

Our first intended solution was using hobbyist IR LEDs hooked into breadboards and powered/controlled by Arduino boards. We spent around $110 total between Arduinos, breadboards, wires, resistors, etc, as well as six 4-inch styrofoam cubic blocks that we figured would make great hosts for the breadboard adhesive pads and fulfill the need for reliable ultrasonic return simultaneously. If the signal block needed to be shortened, we could cut the block down. After some brief testing revealed that the IR LEDs did not broadcast in the direction we thought, we bent the legs at 90* angles to emit horizontally from the tops of the blocks.

The next problem encountered was the emission angle. Our LEDs only emitted in a 30* field of view, and so 12 would be needed, placed extremely carefully, to cover a full 360* of emission. We had only bought 40 emitters, and our breadboards weren’t big enough to comfortably fit all 12 LEDs in a small enough area to reliably cover the full circle.



At a suggestion from Dr. Huber, we began developing an alternative strategy: shining the LEDs straight up onto an angled, reflective surface so that the IR light would be broadcast in 360* that way. We designed solid cones with 45* and 60* slopes in CAD, with 1mm wide holes

in the tips for suspending them above breadboards via rigid wires, and sent them off for 3D printing overnight with the intent of wrapping them in aluminum foil the next day. As we worked on this, the IR seeker problem was fixed (more on that in another section), and so we were able to learn a little bit ahead of time that the IR LEDs weren't strong enough to be detected by the IR seeker at a great enough range, reflected off cones or not.

This necessitated a complete 180 in our strategy, just about a day before the demonstration. We borrowed a very strong IR illuminator from Dr. Huber, and also managed to borrow one from Dr. McMurrough in a late dash to his home. After considering delaying our demonstration for a day to allow for an overnight Amazon order of IR flashlights to arrive (and therefore potentially eliminating the power supply cable issues present in the IR illuminator design we were using), we decided to go for what we knew worked from testing: using these very strong, albeit wired, IR illuminators to shine up at these foil-covered cones, blasting IR light in all directions. To do this, we needed to mount these unstable foil cones centered above the IR illuminators, on a platform that would support them.

Knowing triangles to be the strongest shape, we used 3 strips of Scotch tape (as transparent as possible so as to not block any signal) spaced at 120* intervals to hold the cone, balanced on its point, on the faces of the IR illuminators which had been taped down to roughly-cut-in-half styrofoam cubes. As we finished this, our demo approached, and so we settled with some very brief testing before going in.

Had our IR seeker worked as expected, such last-minute design changes may have been avoidable, leading to more time for testing and refinement, but obviously, things do not always work out so well.

IV. Sensor Issues

As mentioned previously, our IR Seeker sensor was not working as expected. In fact, the EV3 brick did not register the seeker as a sensor whatsoever - only as an unknown i2c device. Complicating the matter was that the seeker had no documentation that we could find anywhere, despite using multiple search engines and advanced queries. The sensor had only been sold from 2007-2011, and so predated most of the resources we had been using.

After trying various solutions made for the V2 sensor, none of which worked, we turned to the ev3dev github page itself and opened an issue. Two days later, a contributor responded to the post, pointing us in the direction of direct i2c interfacing, as he stated that the ev3 would not auto-detect the sensor in any case. This led to another day of learning about i2c, eventually getting to the point where we had written some code that, theoretically, would interface with the sensor. Unfortunately, we got only static readings, no matter what the position/orientation of both the sensor and IR sources.

Consulting with Dr. Huber led us to some semantic improvements in the code, and Dr. Huber found someone's GitHub repository in which they had written an IR Seeker V1 driver that, while not directly giving us a solution, gave us the i2c addresses and registers we needed. Having modified our code accordingly, we got extremely weird and inconsistent strength readings from each of the seeker's five internal readings, and no real directional readings. After over an hour of testing, Dr. Huber tried switching out the seeker with another identical one; lo and behold, our seeker had been broken. Armed with our new direction-sensing functionality, we began testing that revealed the flaws in our signal source block designs, as documented in Part II.

V. During the Demo

During our demonstration, two issues were noticed that our time crunch prevented us from seeing beforehand.

First, in our Love behavior, the robot turned away from the source rather than towards it; we realized this was due to not having updated the code from when we briefly had had our IR seeker inverted, necessitating inverted direction control. Once that was fixed, our Love behavior worked properly.

Second, in our Explore behavior, which we had built off the back of our Fear code, the vehicle would stop when facing away from the source, when it should have instead continued

moving in the direction it was facing. A very quick fix eliminated this bug and the Explore behavior was successful.

Our Aggression and Fear behaviors, polar opposites of each other in behavior but similar in architecture, worked as expected.

We were unable to demonstrate any interesting interactions *between* Braitenberg behaviors, owing both to having just two signal blocks for three vehicles, and an unwillingness to mount unstably-fastened source blocks containing equipment that didn't belong to us onto vehicles that move very quickly and may collide.

VI. Lessons Learned

We learned many things throughout the course of this final project, but we will list just a few here.

1. **Do not assume that "compatible" sensors are, in fact, compatible.**
   Despite these sensors coming with our kits, they were not compatible and did not work as expected. In hindsight, we should have checked basic compatibility and functionality early on; in the future, we will do just that.
2. **Design iteratively.**
   We mapped out our entire design, vehicles and source blocks, prior to beginning work. While doing that is fine, we made mistakes in that when problems were encountered, we did not prepare contingency plans for other things going wrong. When one problem was resolved as the deadline approached, it revealed flaws in other parts of the design. In the future, we will attempt to foresee when a design may have issues, and plan for such issues
3. **Consult wise counsel early and often.**
   Most of our difficulties in this project were the result of waiting too long to consult wise counsel in the form of GitHub communities and Dr. Huber, leading to delays that hindered other parts of our design and compressed available time for final implementation stages. In the future, we will consult with wise counsel early, rather than waiting until we hit true brick walls in our own efforts.

VII. Robot Name

We named our three robots Babylon I, II, and III, after the civilization that ended the rule of the House of David, whose members included Solomon and Rehoboam, over the southern Israelite kingdom of Judah for a period.