CS341
Kenny Hunt
2017/03/29

Mel Gibson

Preliminary

Strategy:

The overall defense strategy was based on two Aikido foundations: Hamni and Tenkan. Hamni is a core ready stance in Aikido, Tenkan is a technique that involves throwing-off another persons balance by “moving out of the way”. The offense strategy was based on mantis-like lifting appendages that are initially oriented low to the ground. So that when opponent has driven on top of the appendages, the appendages spin, changing orientation, and lifting them so that their wheels are no longer touching the ground. After they are unable to move with their wheels we drive them off the edge.

Implementation:

The mechanical construct of the robot went through many different phases until we reached the mantis-like robot. The body consisted of four motors: two `NXT Large Motors`, and two `EV3 Medium Motors`. The `NXT Large Motors` were used for the back two wheels, providing maneuverability around the arena. The `EV3 Medium Motors` were used for spinning the appendages during the `Lift Behavior`. For sensors we used four in total: two `EV3 Touch Sensors`, and two `Color Sensors`. The `EV3 Touch Sensors` were part of the appendage system, being used to check whether or not the appendages were in contact with anything. The `Color Sensors` were placed in the rear to prevent falling off the arena. The software made use of these sensors when performing 4 `Behaviors`: `Hamni`, `TenkanRight`, `TenkanLeft`, and `Lift`.

The `Hamni Behavior` consisted of using the two `EV3 Medium Motors` to spin the appendages in front to be low to the ground, this `Behavior` was always activated at the start of the match. `Hamni` was ordered with least priority in relation to the `Arbitrator Behavior` array.

The TenkanRight and TenkanLeft Behaviors were mirrored counterparts that used the Color Sensors to avoid falling off the arena. These two Behaviors were ordered as the top priority with the Arbitrator, we decided this because we relied heavily on the Touch Sensors to detect any other objects in the arena, which meant that we would have to always be in contact with the opponent robot to know where it is. The TenkanLeft Behavior was activated whenever the Color Sensor on the back right was the first to sense the white border of the arena. When activated, the left wheel's Large Motor was set to a speed of 600 deg/s. This state would continue until the left Color Sensor was sensing the black section of the arena, which would trigger the right wheel's Large Motor to spin backward at 600 deg/s, in an attempt to move out of the attacking robot's way. TenkanRight performed in a mirrored fashion to TenkanLeft, activating whenever the back left ColorSensor was first to sense the white border.

The Lift Behavior was our main offensive Behavior and was ordered after TenkanLeft and TenkanRight's priority. For the Lift Behavior to work, the attacking robot must be on top of our robot's appendages. This state was decided by assuming that if the Touch Sensors are in a state of being “pushed-in/touched” for a long enough period of time, then the opponent robot should be on top of them. Once this was achieved the two Medium Motors would spin orienting themselves to be lifting the other robot off ground (The right Medium Motorspinning 90 degrees clockwise, and the left Medium Motor spinning 90 degrees counter-clockwise). Assuming that the opponent robot's wheels are off the ground our robot would then drive forward off the track, pushing the other robot off first.

Analysis:

During the Preliminary round there were several problems that we encountered, the first was the size of our robot. Our robot's front appendages were too long, so we had remove most of their functionality. Another problem was the 5 second delay, this had not yet been implemented correctly during the preliminaries. The most apparent problem was not knowing when our control software was running, this error was due to the brick placement (trying to be within the size constraints, the brick was placed on the underside of the robot, so the display was not visible when the robot was placed in the arena. This caused several dysfunctional starts by human error: forgetting to press the execution button twice.) The

successes we encountered were often due to greater failures by others. For example, against Cubone, the match was decided by whomever started first. The other would lose simply by not moving as they were pushed off the edge. The true success was found in the crowd's enthusiasm during the initialization of our first match, where we demonstrated Hamni by spinning the frontal appendages: a visual function unique to our robot.

Finals

Strategy:

The main thing that we learned in the preliminaries was that having the brick placed on the underside made for much more inefficient testing (being unable to see what was printed on the LCD). We redesigned the robot attempting to maintain the Lift functionality, but had to scrap it due to size constraints. Deciding on the final robot, we removed the two Touch Sensors and added a single NXT Sonar Sensor (inspired by the performance of other robots that used this effectively in the preliminaries: Sumo and Yoshi). The new offensive strategy became “find the other robot and drive towards it at 900 deg/s”, we kept the TenkanRight and TenkanLeft functionality. The final hardware was: two NXT Large Motors, two NXT Color Sensors, and one NXT Sonar Sensor.

Implementation:

Implementing the the Sonar Sensor was done in the new Magnet Behavior. The Magnet Behavior used the Sonar Sensors distance readings to determine when our robot would attack:

```
while(averageDistanceReading != POSITIVE_INFINITY)
    rightWheel.setSpeed(900)
    leftWheel.setSpeed(900)
```

We replaced the Lift Behavior from the preliminaries with the Magnet Behavior in the array for the Arbitrator so that it had the same priority in offense.

Analysis:

The first problem we encountered in the finals was the balance of our robot. During all the matches against Omnipotent, we lost by tipping backwards off the arena. To prevent this in the next match-up (against TODO) we added a last minute stabilizer to the back of our robot, while this was effective in tip prevention, we still lost against TODO by false start. The single success we had in the finals was again due to having a faster start time than the

opponent robot had (and it is likely that this victory was itself false start that went unnoticed.)

Conclusions:

To improve our robot we would have to build upon the tipping-prevention that was improvised upon the bot during the Finals, we would also have to correct for the false starting time.

I think that the rules were fair, but as mentioned in class, the arena size may benefit from being larger. Another suggestion (although I am not completely sure how this could be accurately regulated) would be measuring and constraining the size in terms of “volume of space being taken up by the robot ”, I believe that this could result in more unique robot functionalities.