Progress Report 10

ENGR 1182: Instr. Busick, GTA Chris Chang
Group D: Blake Harriman, Kyle Kottyan, Kyle Pellikan, Joe Sudar

Backward-Looking Summary

Situation

The items investigated during these performance tests were different coding implementations. This is important because it allows the team to get to a point where the AEV runs the most efficiently and the most precisely. The team tested two different code implementations to see the effects they had on the test runs. After doing various test runs and collecting data, the team could come up with a tentative final code implementation.

Observations & Analysis

The first code implementation the team tested was based off the scenario that the team had been running with for most of the semester. This code focused on having the AEV coast from place to place to maximize energy efficiency. The second implementation of code focused more on precision. This was done through including loops in the code that checked to make sure that the AEV was at the proper location before stopping and starting again. After comparing the two code implementations, the team observed that the implementation focusing on efficiency was more efficient, but much less precise. This is due to the distance the AEV travels will always vary slightly because not all runs are the same. Since there is no code in this implementation to check the position of the AEV, there is no way to make the start and finish point for each task for accurate.

The second implementation of code was found to be slightly less energy efficient, but much more precise than the first implementation of code. The loops in this implementation consider the position of the AEV as it is coasting along, so if the AEV stops short of the gate or of the cargo, then it will start the motors again to push it forward until it reaches the proper position. The fact that the motors need to start again if the AEV is too short when it initially stops causes there to be more energy consumption, but, at the same time, the position the AEV stops at throughout the run is much more precise. This implementation still uses coasting to get around the track, but additionally checks the position of the AEV to ensure that it is where it is and adjusts to correct any mistakes

The team decided that the tradeoff for slightly less energy efficiency in exchange for more precision was worth it, so the tentative final design for the code includes loops that check the AEV's position as to make the landing spots as accurate as possible. This will ensure that the AEV does not fail the tasks during the run which is what ultimately matters for the final run. It doesn't matter how efficient the AEV is if it can't complete the tasks properly.

Figure 1 below is a plot of the energy usage during execution of the two coding scenarios. The second coding implementation that uses a combination of coasting and conditional statements is shown in the figure to use slightly more energy overall than the other scenario, which was solely focused on coasting.

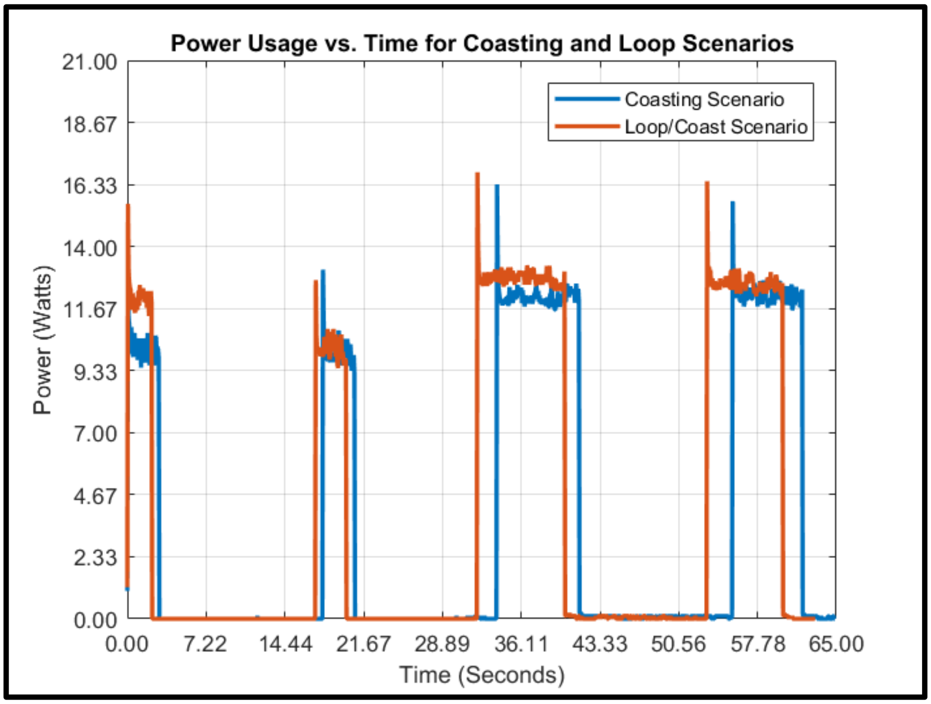

Figure 1: Power usage versus time for the coasting and loop coding scenarios

The graph above shows a plot of the power usage versus time during full runs of the two different coding scenarios tested. The scripts for the runs can be found in the Appendix.

Takeaways

- Precision is more important than energy efficiency
- Coasting is the best method of getting around the track efficiently
- Creating an implementation that eliminates external variables (such as battery power) is vital to the AEV's success

Forward-Looking Summary

The next performance tests are centered around minimizing the total energy usage of the AEV while also fully meeting the requirements outlined in the mission concept review. Two different coding scenarios will be tested and refined to precisely determine which will be the most efficient scenario for the final run in Performance Test 04. With the final design and coding scenario now chosen, the team will make minute changes that will be tested for their effect.

Some of the variations that will be tested during this round of experiments will include varying the initial power at which the AEV is set to make the gliding route more efficient and faster. Although the team is precise in its measurements of the track, the employment of conditional and loop statements will improve this accuracy. Overall, the most prominent concern for the team would be the speed and precision, which will both be worked on during Performance Test 03.

Table 1: Distances Traveled and Time Elapsed by Section

Section of Track	Marks	Wait Time (seconds)
Region 1 (To Gate)	485	0
Gate (Incoming)	0	7
Region 2 (To Cargo)	524	0
Region 3 (Cargo to Gate)	485	5
Gate (Outgoing)	0	7
Region 4 (Return to Start)	524	0

This table breaks down the sections of track that the AEV must travel to complete the MCR into 4 separate regions. For each region, the specific marks that AEV must travel and the wait times necessary were identified. At each gate, the travel distance was zero, though the wait time specified in the table is what is required to activate the sensors and open the gate.

Table 2: Weekly Schedule for the Week of April 27, 2017

Date:	Location:	Time:	Blake H.	Kyle P.	Kyle K.	Joe S.
March 27, 2017	Hitchcock Hall Room 308	1:50 pm-2:45 pm (1 hr.)	PT test 2, wrote code implementing while loops to control precision.	Meeting Note PT 2 testing, u.osu website	Code Data Extraction and analysis	Final edit of PDR, fix issues in magnet connection site
March 29, 2017	Hitchcock Hall Room 308	1:50 pm-2:45 pm (1 hr.)	Edited 2nd code for PT 2	Meeting Note, PT 2 Testing, u.osu website	Code 2, PT 2 testing, Data Extraction and analysis	PT 2 testing, Compare observable difference of each code
April 04, 2017	Houston House: 2nd Floor Lobby	7:00 pm-9:00 pm (2 hrs.)	Backwards Looking Section, final draft of while loop code	Meeting Notes, u.osu website	Appendix, Data analysis, Forward looking	Weekly Schedule, Weekly Goals, Tables

Table 3: Weekly Goals for the Week of March 20, 2017

1	Determine the most energy efficient code implementation
2	Complete Performance Test 3
3	Create a draft of the Oral Presentation
4	Perform constantly perfect runs on the test track

Appendix

Arduino Coding Scenarios

```
// Loop/Coast Coding Implementation
// @author Blake Harriman
// This code uses a combination of while loops and coasting techniques to complete
// its mission.

reverse(4);

motorSpeed(4,35);
goToRelativePosition(130);

motorSpeed(4,0);
goFor(9);

while (getVehiclePostion() < 475) {
motorSpeed(4, 20);
goFor(1);
}

motorSpeed(4,0);
goFor(7);

motorSpeed(4,35);
goToRelativePosition(148);

motorSpeed(4,0);
goFor(9);

while (getVehiclePostion() < 975) {
  motorSpeed(4, 20);
  goFor(1);
}

motorSpeed(4,0);
goFor(6);

reverse(4);

motorSpeed(4,45);
goToRelativePosition(-230);

motorSpeed(4,0);
goFor(9);
```

```
while (getVehiclePostion() > 550) {
  motorSpeed(4, 30);
  goFor(1);
}

motorSpeed(4,0);
goFor(7);

motorSpeed(4,45);
goToRelativePosition(-250);

motorSpeed(4,0);
goFor(9);

while (getVehiclePostion() > 15) {
  motorSpeed(4, 30);
  goFor(1);
}

motorSpeed(4,0);
goFor(8);
```

// Coasting Coding Implementation
// @author Blake Harriman, Kyle Kottyan
// This coding scenario uses solely coasting techniques to complete its
// mission.

```
  // Orient direction correctly
  reverse(4);

  // Start to shutdown point for gliding
  motorSpeed(4,35);
  goToRelativePosition(130);

  // Glide from shutdown position to gate sensor and wait for 7 seconds
  // start to gate takes about 8 seconds, 7 additional for gate opening
  motorSpeed(4,0);
  goFor(15);

  // Run motors at 35% power until relativePosition of 70 marks
  motorSpeed(4,35);
  goToRelativePosition(148);

  // Glide to cargo and wait 5 seconds for secure connection
  motorSpeed(4,0);
  goFor(13);

  // Reverse motors for trip back
```

```
reverse(4);

// Cargo retrieval area to gate sensor
motorSpeed(4,45);
goToRelativePosition(-230);

// Glide to gate and wait 7 seconds
motorSpeed(4,0);
goFor(14);

// 2nd gate glide to end
motorSpeed(4,45);
goToRelativePosition(-250);
```

Meeting Notes

Meeting 15: March 27, 1:50 PM, Hitchcock Hall 308		
Team Members:	In Attendance:	Job/Responsibility:
Blake Harriman	**X**	Coding, Testing
Kyle Kottyan	**X**	Coding, Testing
Kyle Pellikan	**X**	Meeting Notes, Project Portfolio
Joe Sudar	**X**	Scheduling, AEV Maintenance

The team completed the PDR, as well as tested the AEV on the track.

Goals for next meeting:

- Work towards creating more efficient code.

Summary:

- Team tested on the track multiple times, but improvements could still be made.

Notes:

- Team asked questions to see what improvements could be made to the PDR. Then it was submitted.

Meeting 16: March 29, 1:50 PM, Hitchcock Hall 308		
Team Members:	In Attendance:	Job/Responsibility:
Blake Harriman	**X**	Coding, Testing

Kyle Kottyan	**X**	Coding, Testing
Kyle Pellikan	**X**	Meeting Notes, Project Portfolio
Joe Sudar	**X**	Scheduling, AEV Maintenance

The team tested the AEV on the track, and then tested a new code implementation. They then compared the data to the original code.

Goals for next meeting:

- Complete the Progress Report

Summary:

- Ran a new code implementation and gathered data from it.

Notes:

-

Meeting 17: April 4, 9:00 PM, Houston House 2nd Floor		
Team Members:	In Attendance:	Job/Responsibility:
Blake Harriman	**X**	Coding, Progress Report Writer
Kyle Kottyan	**X**	Coding, Progress Report Overseer
Kyle Pellikan	**X**	Meeting Notes,Progress Report Editor
Joe Sudar	**X**	Scheduling, Progress Report Editor

The team got together in order to compete the Progress Report.

Goals for next meeting:

- Finish the Oral Presentation Draft

Summary:

- Team worked on the progress report and reflected on what was accomplished during this lab.