YORK College
THE CITY UNIVERSITY OF NEW YORK

Team Members:

Jonathan Chery

Abdoul Diallo

Wangchuk Dorjee

Samuel Graham

Faculty Advisor:

Daniel Phelps

I have approved this paper

X_____

NASA Swarmathon 2016 Technical Report

The York College Hivemind

I. Abstract

The NASA's Swarmathon Competition is a great competition that amended many students' and participants' teamwork and programming skills. The York College Hiveminds, during the competition, struggled many times because of the lack of resources that we had and the unpreparedness that we experienced. Nonetheless, we were still able to participate in the competition with the determination to succeed. The competition was more of a learning experience for us to work on in the future rather than a competition to win. With that being said, we hoped to improve and add on to the programming of the Swarmies for the future.

II. Introduction

During the NASA's Swarmathon Competition, the York College Hivemind had some trouble with understanding the code and implementing or amending new algorithms or codes to the existing one. Also, with us approaching the competition late, we weren't able to attack this competition in the way we wanted to because of the time. We also had some trouble with agreeing on times to meet or to come together to approach the competition and to gather more individuals to join our team. However, with what we had, we were able to manage and do our best in participating in the competition.

Through all the trials and errors, we eventually came to agree on what searching algorithm we would like to implement onto the Swarmies. Also, with our lack of resources, we were only able to focus on the mobility part of the code in which we tried to make the Swarmies spiral around until it finds an element. Once the element was found, the Swarmies would go back to the base, and start again from the beginning. The spiral algorithm was a start for us and our main focus was to implement the code within the mobility aspect of the Swarmies. We've obtained this idea from our faculty member, Daniel Phelps, in which he told us to work on making the Swarmies go around in a spiral motion. This idea was difficult since we did not necessarily know how to approach it but, we were able to make the Swarmies move around in a circular motion. This was a start for the York College Hivemind because it gave us an idea of how to approach the mobility aspect of the Swarmies.

III. Related Work

While doing research on similar algorithm, we were able to find a work that was similar to our approach in the competition. This work was from a youtube page by Matthew Fricke. He posted many searching algorithms using swarms of robots that we found interesting. The algorithm that truly interest us was an algorithm that was similar to the spiral algorithm we wanted to implement. The swarms of robots were moving in a rectangular spiral staircase motion and collecting elements to bring back to the base. Once brought, it would start again until it finds a new element. This caught our attention because it was the algorithm that we had in mind to implement but it was different because we weren't planning on doing a rectangular spiral staircase but a spiral staircase. Attached is a link to the related work that we were following: https://youtu.be/qeiGRDlfsIc.

III. Methods

For this research, we had multiple algorithms we wanted to implement in order to make the search time more efficient, but due to the time restriction we had, we decided to go with a spiral staircase algorithm. The spiral staircase algorithm, is an algorithm which makes the rovers turn in circles from the center outward increasing the area in which they turn by a specified amount. While performing this algorithm, if any of the rover finds a tag, it would return to the center, register the found tag, then go back to the original location where it found the tag. If there are other tags at the same location, this step is repeated until there are no more tags left. This was the best approach because then the robots would cover the entire area synchronously reducing the chance of a tag not being discovered. Another approach we tried was to improve the random search algorithm which came with the rovers. The goal of the improvement is to make the rover go back to the original location if it found more than one tag.

Our main focus was trying to implement the the spiral staircase algorithm into the mobility aspect of the code. This pose rather difficult due to the fact that we did not necessarily know where to start. However, as time passed, we were able to start with getting the Swarmies to move in a circular motion. We first implemented the code:

```
int main(int argc, char **argv)
{
gethostname(host, sizeof (host));
        string hostname(host);

if (argc >= 2) {
publishedName = argv[1];
cout << "Welcome to the world of tomorrow
" << publishedName << "!  Mobility module
started." << endl;
        } else {
        publishedName = hostname;
        cout << "No Name Selected. Default
is: " << publishedName << endl;
        }

ros::init(argc, argc, "round");// This creates
the node of the programe
ros::NodeHandle nH;
round_publisher =
nH.advertise<geometry_msg::Twist>((publi
shedName + "/velocity"), 10)//this

        moveRobot(0.2, 0.0, 0.5);
        ros::spin();

        return 0;
}

// This Function will Move the Robot in a
circular motion

void moveRobot( double x, double y, double
z)// Distance how far to move
{
        velocity.linear.x = x;
        velocity.linear.y = y;
        velocity.angular.x = 0;
        angular.angular.y = 0;
        velocity.angular.z = z;


// now we publish the.results
        round_publisher.publish(velocity);

}
```

This code was able to make the Swarmies move around in a circular motion however, it wasn't in a way we had in mind. It only made the Swarmies move in a circle. We attempted to fix it but our approach to fixing it did not result as expected. This brought many issues to our approach to the competition because we were stuck on improving the code we have written for quite some time.

We thought that a good approach to solving this issue was to increase the number by 1 or by a fraction greater than zero(0) but less than one (1) for every time it made a circle. This idea soon posed to be difficult to implement because we had to figure out a way to indicate to the robot that it completed a circle through its motion.

In conclusion, we focused more on trying to look for a way to implement the spiral algorithm but was confused on approaching the task. It wasn't an easy task for us to make the Swarmies move in a spiral motion however, we had a start, which was making the Swarmies make a circle, which helped us to better understand the code given by NASA.

IV. Experiments

Due to the fact that we only focused on the mobility aspect of the code, the experiment that we ran were minor and weren't all sufficient. It took us some time to test the first algorithm because we did not know how to implement the code. However, after some time, we were able to implement the code. We only did a few experiment in the GUI and it worked well enough for us to feel comfortable with uploading the code that made the Swarmies move in a circle.

We wished we had experimented more often so we can better the original code but team problems came along the way that hindered us to do so. The experiment we did, did help us better understand how the Swarm environment worked and gave us the opportunity to focus on key aspect of the code provided by NASA. From our experiment, we were able to find what mobility code we needed to improve on.

V. Results

We were disappointed by the results we found because it wasn't what we expected. We expected to have a better approach to the competition, yet alone to the searching algorithm, however, this did not happen. We weren't able to gather any result on our code nor findings. We had many difficulties getting the code to work and uploading it to the Github repository. Another problem we came to was understanding the code NASA provided. The fact that we had a lack of resources caused us to struggle with the competition.

The code we were able to add, we were proud of and hope to add and improve on it. Our goal was to create a spiral staircase algorithm but, we struggled with doing so. So we only stayed with making the Swarmies move in a circle and hope to amend it in the future. Therefore, with the difficulties and struggles we've gone through, we weren't able to find any results. In conclusion, we were unprepared for such challenge however, we still tried our best to attack the code and participate in the competition. We hoped for the next few years, we will be better prepared and have a better approach to the competition.

VI. Conclusion

This competition was well needed. We were able to improve our technical along with teamwork skills. However, we weren't able to do much. We've struggled the majority of the time with implementing new algorithms and improving the original code that were given. We had an idea of what we wanted to do but, we've struggled in implementing the idea. We knew we wanted to make the Swarmies find elements using a spiral staircase algorithm in which they would spiral in a staircase motion until they found an element and, once found, bring back to the home base. Therefore, we only worked on the mobility aspect of the

Swarmies. With only a three-man team, we weren't able to gain the result we wished we had found. Our main problem was our lack of resources and our understanding of the code NASA provided. We hoped that in the future, we will be able to improve on the code we worked on and become better prepared for the next competition.

VII. References

We only used two(2) references which were Matthew Fricke's youtube videos and "Programming Robots with ROS" by Morgan Quigley, Brian Gerkey, and William D. Smart.