Winston-Salem State University

Development of WSSU Rams Swarm Algorithm for NASA's Swarmathon Competition

Joel Evans, Lisa Hauser, Michaelangelo Fields, Martine

ABSTRACT

This paper presents the research, architecture and program development of swarm robot behavior for the prospective exploration of Mars. It also describes in detail the methodology that we have implemented to assist us with solving the Swarmathon simulation. Many hurdles have been hit, some overcomeable others we fail to overcome, during this learning process for the team. The final goal of the algorithm was to increase rover productivity by working cohesively with one another, however our implementation of the algorithm has not work and we are able to test the effectiveness of our algorithm .

Keywords - assembly line, swarm robots, divide and conquer, swarm controllers, architecture, algorithm

II. INTRODUCTION

The “swarm” term has been used to identify different types of system in engineering and computer science. This type of robotic science has heightened because of it use to make new discoveries in space. The advantages of this systems are robust, adaptable, scalable, inexpensive and efficient. The Swarmathon is a competition developed by the National Aeronautics and Space Administration (NASA) for multiple universities across the United States to develop a unique search algorithm for a set of swarm rovers to seek, collect, and return tags around an designated area. This is meant to innovate the development of swarm algorithms for use in space exploration and collection.

As a group a few obstacles had to be kinked out before we had the chance to work on the actual coding aspect. The first thing we had to sort our was meeting times for the team. This task wasn’t too complicated, but it did have some slight difficulties because of the extremely varying schedules team members had. Followed by this we need to come up with an algorithm for the robots to search before touching the code. Multiple ideas were tested on a whiteboard and brought to the table, until we decide on what we thought was the best fit algorithm. After this things did began to become a little shaky. Unfortunately due to delays and I the lack of Ubuntu on our personal machines we had a lot start in getting the actual system installed. Once this was done we needed to dive into the simulation system and some of the code. We were able to figure out the simulation and understand how it operates and the various sensors the robots use to operate and navigate. The biggest hurdle came when approaching the code. The first difficulty came because of the team's lack of understanding in C++. We had a few

people who could understand it really well, but weren't as fluent in writing in it as they were with other languages. The biggest issue came from implementing, as well as appending new algorithms to the existing code. Our ideas just didn't seem to mesh well with what we wanted to accomplish.

III. RELATED WORK

A few of our original ideas for our original ideals came from video we saw on Youtube and other reading previous papers for the competition. However, we soon realized that we didn't really like any of the algorithms we had came across. They either seemed not efficient enough, or stepped beyond our cophersion. We decided that it was best for us as a team to naturally brainstorm and come up with an original algorithm that we could use in this competition.

IV. METHODOLOGY

Upon observing the technical reports as well as built in algorithms we have brainstormed many ideas that would increase rover productivity. The algorithm that we have created mimics an assembly line. We decided that we wanted to first split the map up into three separate pieces, each piece overlapping the other (Note we planned on doing three pieces for the primary rounds and increasing the number to size once we got the the final rounds). We also didn't want to make these pieces too big. This is why we overlapped them and wanted to maintain a distance that wasn't too far from the home. Much like what we envisioned a space exploration would be we astronauts wouldn't go too far from the base, it would take too much time and resources to search remote locations. The same things apply to the rovers. At the rate in which they move it would take too long for them to scan the outer regions of the map and the tags bring back to the center

.

Like mentioned before will have each robot scan and search for tags in a specific area. Once it finds a tag based on its region's proximity from the circle of the map, the drop off zone, it passes that tag along to another robots region. This region that it passes to is what we call the discovery zone. While a robot is constantly searching its region it will periodically check the discovery zone. The reason being is because there might be tags that are easily located here for them. This is where the assembly line idea takes place. Tags are passed from rover to rover until it reaches a rover that is close to the drop off zone. His job is to then take all tags there.Each rover has its own sub task which helps the overall goal get accomplished. We also wanted

thier to be some type of communication to happen with robots to help identify when they should go to their discovery zones. The thought process behind this was having the robots use the camera sensor and sonar sensor to help. The Came would be used as eyes to see tags in the discovery zone or a robot heading in the direction of the discovery zone. While the sonar is used as a tool to tell close robots to check the discovery zone. A robot would cross the path of the other three times in order to signal that the other robot should go view waht is located in their discovery zone. This approach only works when they are close to each other, so sticking to a routine search in the discovery zone makes the process easier. Note that all robots don't have a discovery zone. For example the outermost reaches have a prime function of just searching and dropping off tags to other robots.

In code this would work by a rover finding a tag. It would then pass the tag over to the discovery zone which much like the center of the map has it's own coordinates. The position of the tags essentially kept in a temp variable. The temp variable is then passed to another rover to retrieve, or the rover will recover it by skimming over its discovery zone. One rover is only responsible for scanning the outermost premises for the preliminary round as we gather more rovers we will have a bigger area to cover for each robot scanners.

The retrieve method will go as follows: temp variables for the retrieval rovers. The scanner rover will send the temp variables to the two retrieval rovers. Every other tag will be sent to temp2 variable and vice versa. The robots will then take the tags and unload them onto the center drop off zone. The purpose of unloading the tags on the center line and keeping a rover close to the drop off zone is to minimize the time it takes for the rovers to travel. The rover will then have an unloading method after every 5 tags, or so have been retrieved and transfer to the drop off zone the rovers return to the position of their last scan, before they got to the discovery zone and continue to search in their regions.

V. EXPERIMENTS

Due to the fact that our code did not work they way we had intended the algorithm to our experiments were extremely limited. We were able to get the rovers to search randomly in space, but in practice it proved to be extremely hard to split the map up into multiple parts. This was the first hurdle that we saw when trying to implement this algorithm. The next hurdle was trying to get communication for the robots. With our lack of understanig in C++ and the whole ROS system we couldn't find a way for these

robots to send information to each other. We are sure there is some way to achieve this however we were unable to crack this problem. With these two issues in mind we thought we could be able to implement the rest of the algorithm. This proved be another difficult task because we lacked the underlying foundation of our algorithm which was communication. However with this still missing we could still have the robots on a time schedule that would let them know when they should go to the discovery zones. This issue with this was editing the code so that the robots moved in a more uniform manner. Due to time constraints we are unable to figure out exactly how this can be accomplished. One idea stemmed from sending each robot out at different times to their regions. And having them have a timer start once each robot made it to its region. The ones with discovery zones would then stop at the 5 minute mark and go check their discovery zones.

VI. RESULTS

The results for our experiment have been minimal this is based off the fact that we couldn't get the code to work the way we intend it to. Comparing and conThe code that we have doesn't fulfill the need of searching and retrieving tags. The code given to us by default, that searches in random, on average yielded for tags than we did. This is quite unfortunate and disappointing considering the speed of that randomized searching is very slow and inefficient compared to other algorithms researched on the internet.

VII. CONCLUSION

In conclusion this competition was needed. As a team we had the chance to jail and problem solve as a unit. This is an essential skill for any software developer in the real world. Most projects will be talked as a unit and it great to get accustomed and learn to build cohesion on the go. Our team did not execute and implement our algorithm too well. We overall struggled with the project at hand. Our lack of resources, understanding of the code and how the system worked was a huge flaw we had. This was our first year experimenting with the system and it is evident that we needed this year to get acquainted with how the rovers operate, and how the whole simulation system works. Next year different implementation will be used and we will actually have a better foundation on what can and what can’t be done. Again the team viewed this as a buffer year to really under ROS and the power behind it. We plan on coming back next year doing exponentially better.

VIII. RESOURCES

No resources were used to write this essay.