

CS CAPSTONE PROBLEM STATEMENT

OCTOBER 19, 2017

NASA UNIVERSITY STUDENT LAUNCH INITIATIVE

PREPARED FOR

MECHANICAL ENGINEERING, OSU

DR. NANCY SQUIRES

PREPARED BY

GROUP 33

CODE MONKEYS IN SPACE

MARK BEREZA

JOSEPH STRUPH

KEVIN TURKINGTON

Abstract

The purpose of NASA's USLI is to construct and launch a rocket that will go at least a mile above ground, safely land, and deploy a rover capable of autonomous movement that will deploy solar cells after moving at least 5 feet from the rocket. In particular, the CS students on OSU's USLI team are responsible for designing, implementing, and testing all software necessary to accomplish this task. This will include rover motor control and obstacle avoidance, graphical representation of test flight data, and creation/maintenance of a website hosting project information and deliverables.

CONTENTS

1	Problem Definition	2
2	Proposed Solution	2
2.1	Website Creation/Maintenance	3
2.2	Rover Intelligence	3
2.3	Rocket Flight Information	3
2.4	Outreach	4
3	Performance Metrics	4
	References	5

1 PROBLEM DEFINITION

The problem posed to the CS students taking part in OSU's USLI team is a small but important part of a much larger endeavor. The Oregon State University USLI team has chosen the deployable rover experiment, defined in the NASA USLI Handbook as follows:

Teams will design a custom rover that will deploy from the internal structure of the launch vehicle. At landing, the team will remotely activate a trigger to deploy the rover from the rocket. After deployment, the rover will autonomously move at least 5 ft. (in any direction) from the launch vehicle. Once the rover has reached its final destination, it will deploy a set of foldable solar cell panels [1].

Overall, the goal of the competition is to:

- 1) Construct and launch a rocket carrying a payload at least a mile (5,280 feet) above ground
- 2) Have the rocket deploy a parachute and safely land within 2,500 feet of the launch point
- 3) After landing and after a button press, deploy a rover
- 4) Have the rover drive autonomously at least 5 feet away from the rocket landing site
- 5) Have the rover deploy solar cells after being at least 5 feet from the landing site
- 6) The solar cells will increase in surface area after deployment (unfold)
- 7) Create formal technical reports regarding rocket/payload design and present them to a NASA review panel during several formal design review meetings
- 8) Participate in educational outreach and get at least 200 individuals involved in the project
- 9) Maintain a website detailing project information and hosting all competition deliverables

The subset of these overall goals that pertain to the CS capstone students involves the research, design, implementation, and testing of software for the rover, in-flight rocket positional information, and the aforementioned website. Additionally, the CS capstone students must contribute software design specifications and research to all reports and presentations required by the competition. These include:

- 1) Preliminary Design Review report & presentation
- 2) Critical Design Review report & presentation
- 3) Flight Readiness Review report & presentation
- 4) Launch Readiness Review outline
- 5) Post-Launch Assessment Review outline

Finally, the team as whole, CS students included, must engage at least 200 participants by the end of the competition through direct, instructional, hands-on activities that serve to educate students in a STEM topic relating to the project. Educational Outreach is involved in this project not just as something NASA would like to promote, but as an actual category for scoring the competition as well.

2 PROPOSED SOLUTION

Each proposed solution to the aforementioned problems can be separated into one of four categories: website creation/maintenance, rover intelligence, rocket flight information, and outreach.

2.1 Website Creation/Maintenance

The website will be fairly minimalist in appearance and hosted on the custom domain osuusli.com. Most of the content will be contained on the home page and the site itself will be implemented using a template. The site will contain, at a minimum:

- The team name and logo
- A list of all participants and their roles in the project
- A brief description of the project and our goals
- Download links for all project deliverables in .PDF form
- Download links for important design documents
- A visual time line of important events with dates, locations, and descriptions

Additionally, the website will be regularly updated as new information comes to light, membership grows, and deadlines come and go.

2.2 Rover Intelligence

In order to accomplish the broad goal of making the rover move autonomously, avoid obstacles, and deploy solar cells, the rover's micro controller must be loaded with fairly sophisticated software. Software for the rover will need to be capable of SLAM or simultaneous localization and mapping to track the location of the rover and properly navigate the landing area.

In order to avoid the ever-present software development issue of reinventing the wheel, the Robot Operating System framework will be used to design software for the rover's intelligence. The Robot Operating System provides extensive tutorials and libraries in C++, Python, and Lisp specifically designed for robotics applications, making it an ideal tool for our purposes.

As for the programming language used for rover software, C++ will be used due to its high performance, low memory overhead, its extensive standard libraries, and its support for multi-processing (a feature missing from most Python implementations). The micro controller for the rover will be a Raspberry Pi 3 running Raspian, a Linux distribution specifically designed to run on a Raspberry Pi with support for ROS, making it the obvious choice for our purposes.

Software will be written for the rover that will allow it to read data from its radar sensors and construct occupancy grids. These occupancy grids will be combined with a software-implemented movement algorithm to perform autonomous movement aided with obstacle avoidance. Additionally, software will be designed to use sonar sensor data to determine the absolute distance and angle of the rocket relative to rover, which will allow it move away from the rocket and determine when it is at least 5 feet away. Finally, software will be designed for the rover to eject the solar cells once it has determined it has sufficiently cleared the landing site.

2.3 Rocket Flight Information

The rocket will be equipped with, at a minimum, an altimeter for elevation measurements and a GPS sensor. Information gathered by these sensors will be stored on a data logger during all test flights, and information from this data logger must be extracted and displayed in a human-readable fashion via a graphic interface. A mature, easy-to-write programming language is a proper fit for designing a graphic interface such as this, so either C# or Python will be used.

2.4 Outreach

The goal of reaching 200 people over the course of the competition will be reached through a combination of various strategies. While the team will try to meet and exceed this number, the quality and intent behind the outreach is important too. Outside of the K-12 requirement, the project sponsor would like to foster continued interest in projects like this at OSU from the AIAA (The American Institute of Aeronautics and Astronautics) club and students in the College of Mechanical Engineering. At the moment, these include leveraging primary school contacts to schedule classroom demos and presentations, distributing fliers, and collaborating with OSU clubs like rocketry and robotics to spread the word about our amazing project. Additionally, each member of the core team will volunteer to take an active role in educational outreach for two separate months between now and the competition's conclusion.

3 PERFORMANCE METRICS

The following performance metrics will help evaluate the success or failure of this undertaking:

- The website runs without errors and is publicly accessible
- Users of the website can download PDFs for all deliverables before the due date.
- The rover will operate autonomously after the single button press for payload deployment
- The rover will move at least 5 feet from the rocket's land site before deploying operational solar cells
- The rover will drive its required distance without getting stuck on any obstacle for more than 15 minutes
- Software will be written that generates an accurate graphical representation of all data gathered by the rocket's data logger
- All software written will be maintained using Git version control with remote repositories hosted on GitHub
- All software changes to the master branch will only be made after review and approval from at least one team member
- All code must compile without errors or warnings before deployment to the rover
- All software written for the rover will follow a single coding style
- All members of the CS team will attend every NASA review meeting
- The CS team members will contribute content to the PDR, CDR, and FRR reports describing the software utilized in the project, design decisions, and any pertinent research
- All CS team members will abide by the rules set forth in the launch and safety checklist, the local rocketry club's RSO, the Federal Aviation Administration's laws, and the High Power Rocket Safety Code
- The team as a whole will reach 200 people before the competition's conclusion

REFERENCES

- [1] Sandra May. Student launch handbook — nasa. <https://www.nasa.gov/audience/forstudents/studentlaunch/handbook/index.html>, August 2017. (Accessed on 10/09/2017).