NASA RMC: Lunabotics

Submitted May 6, 2021

Students
David Houbre (ME)
John Mushatt (CS)
Lisa Spalding (RBE)
Shane Villafana (RBE)
Keming Wang (RBE)

Advisors
Prof. Joshua Cuneo (CS)
Prof. Ken Stafford (RBE/ME)
Prof. Therese Smith (CS)
Prof. Michael Ciaraldi (CS)

0.1 Abstract

The NASA Robotic Mining Challenge is a yearly competition in which university teams build a rover to navigate rough terrain and mine and deliver simulated icy regolith. With strict size constraints, additional challenges are to minimize the rover mass, energy used, and communication bandwidth, as well as achieve full autonomy. Our MQP developed an original design for an all-new rover with a rocker-bogie chassis and a novel, two degree of freedom, backhoe-style excavator. We were only able to build and test the excavator portion of this robot due to COVID and time constraints. We also programmed and tested the state machine and navigation system, creating a code base that can be used by future teams.

Contents

0.2 Authorship Page

Section	Author(s)	Editor(s)
0.1 Abstract	Lisa	Advisors
0.2 Authorship	Lisa	Lisa
1 Introduction	Dave	Lisa
2 Background	All	Dave, Shane, John, Lisa
3 Design		
3.1 Mechanical	Shane	Dave, John
3.1.1 Drive Train	Shane	Dave, John
3.1.2 Mining System	Shane	Dave, John
3.1.3 Delivery System	Shane	Dave, John
3.2 Electrical	Lisa	John
3.2.1 Actuation	Lisa	John
3.2.2 Sensing	Lisa	John
3.2.3 Network	John	Lisa
3.3 Software	John	Lisa
3.3.1 Hardware Abstraction	John	John
3.3.2 State Machine	John	Lisa
3.3.3 Navigation Algorithm	Lisa	John
4 Methodology and Testing		
4.1 Excavator Construction	Shane	Lisa, John
4.2 Excavator Testing	Dave	Lisa, John
4.3 Software Build	John	Lisa, John
4.4 State Machine	John	Lisa, John
4.5 Navigation Code	Lisa	John
5 Analysis		
5.1 Force Calculations	Keming	Lisa
5.2 Gear Calculations	Shane, Lisa	Shane
5.3 Electrical Calculations	Lisa	Shane
5.4 Excavator Performance	Dave	Lisa, Shane
5.5 Software Performance	John	Lisa
6 Project Organization	Lisa	Shane
6.1 Project Tools	Lisa	Shane
6.2 Impact of COVID-19	Lisa	Shane
7 Social Implications		
7.1 Robot Safety	Lisa	Dave
7.2 Imps of Lunar Mining	Lisa	Dave
8 Conclusion	Shane	Lisa

1 Introduction

Ever since NASA was formed 63 years ago, it has led the way in science and engineering. The NASA Robot Mining Competition was designed to inspire the creation of new and innovative robots capable of navigating and excavating autonomously on the Moon's surface. It was the goal of this team to design, analyse, construct and test a robot that would meet the competition's expectations. To this end a design was created, mathematically analyzed, and revised. The students then manufactured appropriate parts based on the revised design. The excavator subsystem was built to test that the design was adequate to meet the expectations of the competition and would work in reality. This paper details the process taken to develop a working robot and shows how this design can be built upon and used for future WPI teams participating in the NASA Robot Mining Competition.

2 Background

2.1 The Competition

The NASA Lunabotics Robotic MiningCompetition (RMC) is a competition in which teams of students from universities all around the United States and internationally design and build robots capable of mining Lunar material and safely delivering the material to a payload site [1]. NASA has been running this competition since 2010. In recent years NASA has shifted the competition's focus from mining on the surface of Mars to mining on the surface of the Moon. This shift was driven by the U.S.'s goal of sending humans back to the Moon in order to establish sustainable exploration on off-world terrain [1]. This Lunar exploration will be used as a stepping stone for larger space missions from the Moon to Mars and beyond [1].

Figure 1: Robot Team at Competition [1]

Although the goal of the project is to create a robot capable of mining materials on the Moon, the robot will not actually be used on the Moon. The purpose of the competition is two-fold. First, by hosting this competition, NASA is able to gain valuable data and concepts on excavation designs, robot autonomy and surface motion operation to help future excavation missions on the Moon. Second, it gives students a chance to hone their engineering skills and apply what they have learned in universities to real world applications. The engineering skills developed throughout the course of this competition will be valuable no matter what field the students decide to pursue after graduation. Overall the RMC is a valuable learning experience for all of those involved.

The winner of the competition depends on who scores the most points during the mining process [1]. For the competition the robot will need to navigate across a simulated Lunar landscape to a designated mining area all while managing randomly placed obstacles in its path [1]. Once in the mining area, the robot needs to excavate material that replicates the lunar surface. This material is made of a simulated basaltic regolith (BP-1) and an ice simulant (gravel) [1]. Once the material has been excavated the robot needs to navigate back to the starting area where there is a station for the gravel to be dumped. The robot

Figure 2: Indoor mining area. [1]

has 15 minutes to deliver as much gravel as possible. Since the beginning of the RMC in 2010, the rules have been slightly changed based on the elements of the robot that NASA finds most important [1]. It is clear from the rubric provided to the students that NASA believes there are three elements of a Lunar robot that are crucial for missions on the moon. These elements are the robot's mass, the amount of gravel mined and the autonomy of the robot [1]. It is the challenge of each team to determine how they should focus their time and resources on coming up with a design that can satisfy these three major categories.

2.2 Previous RMC Robots

The students of the WPI Lunabotics team designed and built a robot to compete in this year's RMC. This year's team began by taking a look at previous years' trials as well as examined and experimented with new ideas in an attempt to develop a reliable and efficient robot for the competition.

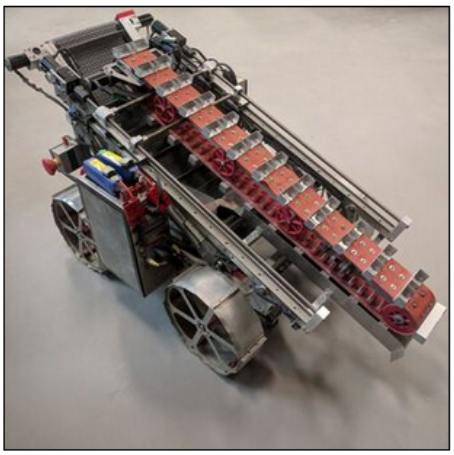

Figure 3: University of Alabama's 2019 Competition Rover [2]

The first team we examined was the University of Alabama because they have won the past five competitions in a row. The team has many things contributing to their success. The first is they have a very large RMC program with a team size of about 38 members. This allows the team to win most of the non-competition awards for their presentation and demonstration, engineering paper, and the public outreach. The second is that they slowly improve their design over time. Rather than starting from scratch, the University of Alabama will focus their efforts on a few parts to make them the best they could be. This works because it allows them to skip the testing of other areas that they know already work. Currently, the team uses a popular design of a dual conveyor belt system: one for collecting and one for dumping (Figure 3). Many teams have used this same system with a few notable variations.

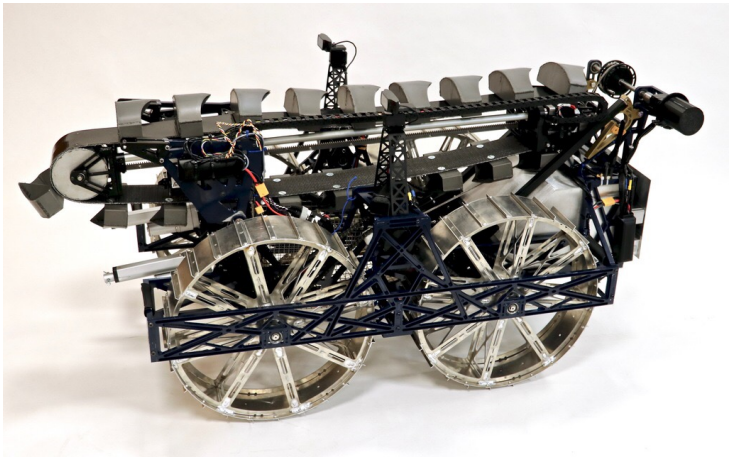

Figure 4: Case Western Reserve University's 2019 Competition Rover [3]

Another team that caught our eye was Case Western Reserve University. Following the same basic design as above, this team was able to collect more than ten times the amount as the University of Alabama. Unfortunately, NASA did not hold the physical competition for 2019, and their impressive performance was not counted. As seen in Figure 4, the main difference of their design was their scoop size. Digging through gravel can be challenging, but this team implemented a design that worked better than any other team on the simulated field. Again, this university had a very large team size and many sponsors, comparable to the University of Alabama.

Finally, a robot that our team found very intriguing was a robot designed by a team from Oakton Community College. This robot features an excavator system made of a simple four bar mechanism. A four bar mechanism, or four bar linkage, is a simple movable closed chain linkage. It consists of four links connected in a loop by four joints. This allows the links to move in a single plane and can be manipulated to create the desired movement of objects. When fully retracted the entire excavator subsystem lies within the dimensions of the robot. With a simple rotation of the crank, the digging scoop which is located at the end of the coupler, is in a position ready to dig. A pair of linear actuators are then used

to push the crank of a smaller four bar which is used to actuate the scoop. With just two rotating members, the excavator is capable of digging up regolith and dumping it within the confines of the robot. This simplistic design inspired the WPI team to create their own, more improved four bar excavator.

Figure 5: Oakton Community College's 2018 Competition Rover [4]

2.3 Previous WPI RMC Robots

Previous WPI RMC teams tried a few different designs. In 2016 a Troy-Bilt snowblower was converted into a mining robot. The snowblowers auger was effective at digging up the regolith, but the team found that the auger was very heavy and didn't dig up as much material as they had hoped [5]. In 2017 a robot called MARKHOR was designed to use a bucket ladder system. This system proved very effective at collecting regolith, but it also kicked up a lot of dust which could damage the chains over time [6]. MARKHOR's success in digging up a lot of regolith undoubtedly inspired the next couple years of RMC design at WPI. The teams from 2018, 2019 and 2020 all used a bucket ladder excavation system [7] [8] [9]. Although this year's WPI team appreciated the ingenuity and success of previous teams at WPI, we ultimately decided to come up with an entirely new design. This decision was based on a few different factors. As previously mentioned, bucket ladder systems tend to kick up a lot of dust, and they also draw a lot of power. While excellent for surface mining, they seem to struggle and get caught on icy regolith when digging deeper. The team decided to avoid these problems and go with a simpler, more efficient design that would minimize dust and the power necessary to dig.

3 Design

3.1 Mechanical

To begin designing the mining robot, the team researched mining techniques, lunar rovers, and successful RMC robots. The team decided to design a four bar excavator with a rocker-bogie drivetrain (Figure 6). This design was chosen for the simplicity of the four-bar excavator, the ability of the rocker-bogie to traverse over obstacles, its ability to mine large amounts of gravel, and its uniqueness among previous RMC robots.

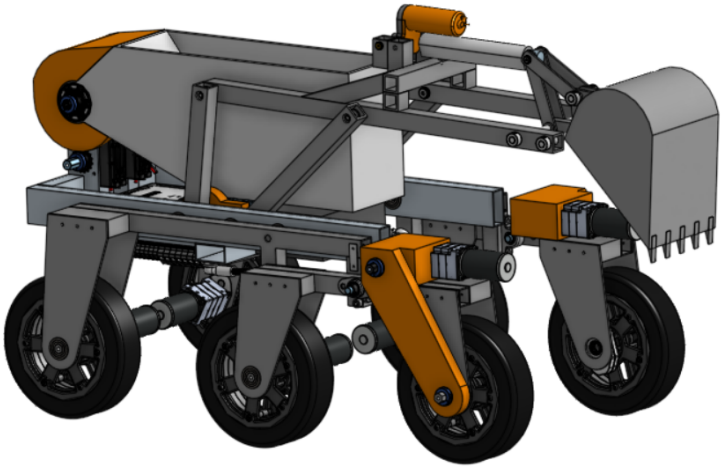

Figure 6: Full design

3.1.1 Drive Train

With the total number of possible boulders and craters being unknown [1], the team wanted a drive train that could travel over smaller obstacles if needed. To accomplish this, the team chose to implement a rocker-bogie suspension (Figure 7). This comprises a rocker, bogie, and differential. The rocker is attached to one wheel, the chassis, and the bogie. The bogie connects two wheels to the end of the rocker. The differential bar pivots at the center of the chassis and links to the rockers on each side. This system is widely used on NASA rovers but is not a popular design in competition. This is partly because fixed wheels are a sufficient solution if all obstacles are avoided. However, it is extremely common for robots to create their own craters while digging. Taking this into consideration, the team wanted to proceed with the more mechanically complex but adaptable system.

The advantage of this type of system is its ability to drive over peaks and valleys with every wheel touching the ground. This type of system also gives the robot the ability to drive over boulders that are twice the diameter of the wheels (Figure 8). For this competition,

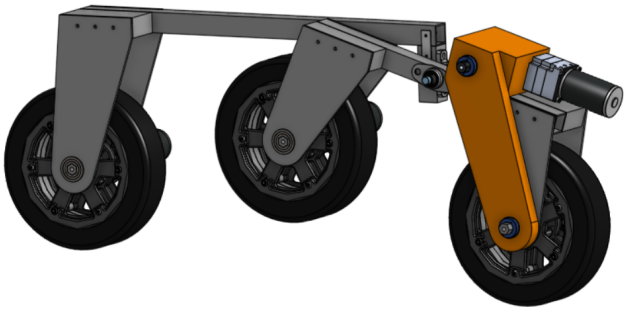

Figure 7: Rocker (left) and Bogie (right)

rocks have an average diameter of 40 cm. This influenced the decision to have a wheel size of 20.32 cm (8 in) for a worst case scenario.

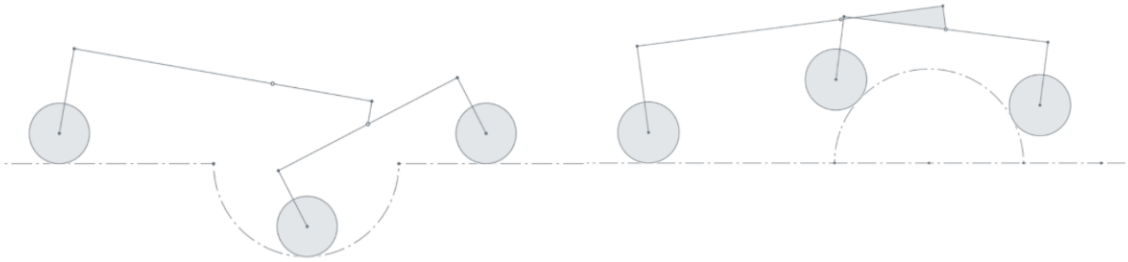

Figure 8: Sketch of Rocker-Bogie driving over a crater and boulder

The wheels themselves are designed with a Vex rim and a custom 3D printed tire to achieve the desired wheel diameter. Each wheel is individually powered with the back four being driven directly and the front two implementing a chain and sprocket. This configuration allows the four bar excavator to dig without interfering with the motors. 3D printed covers are added to prevent dust from compromising the system. Because of the PB-1 simulant that the robot will be driving on, a slow constant speed is desired for moving on the field. This removes the need for spring suspension on the individual wheels and they can be rigidly attached to the rocker-bogie frame. This is done using 0.125 in aluminum plates. The frame of the rocker and bogie are made using 1 inch square aluminum tubing with 1/16 inch wall thickness. This tubing allows for a strong yet lightweight frame.

Another advantage of this system is the chassis only moves half the rotation of the rocker. This keeps the chassis from tipping and spilling the payload collected after excavating. The differential bar is how this motion is accomplished. The bar is made from a 1/8th inch aluminum plate (Figure 9) and is connected to the rocker via ball and socket joints. Because

the bending moment acting upon the differential bar is greatest in the center, it is designed to be wider there for strength.

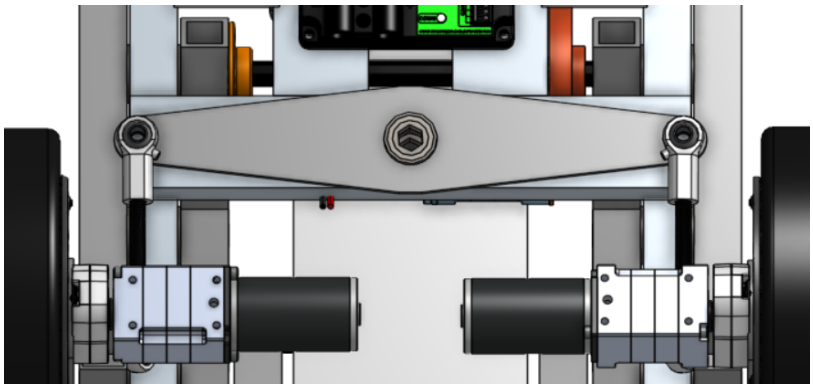

Figure 9: Differential Bar

3.1.2 Mining System

The mining plan is to quickly move large amounts of material. This is accomplished using a four bar linkage and an excavator scoop (Figure 10). The reason for using a four bar linkage is that it is a simple system. This results in it being easy to actuate, and there are fewer parts that can break. The linkages are made from 1 inch square tubing with 1/16 inch thick walls.

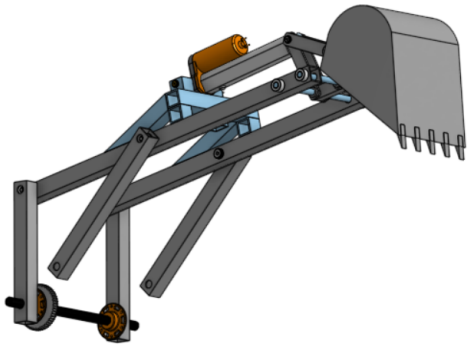

Figure 10: Excavator Four Bar

The scoop is made out of 1/8 inch thick aluminum plates and is actuated with a smaller four bar linkage (Figure 11). It can fit roughtly 5-6kg of material and reaches about 40 cm deep when fully extended downward.

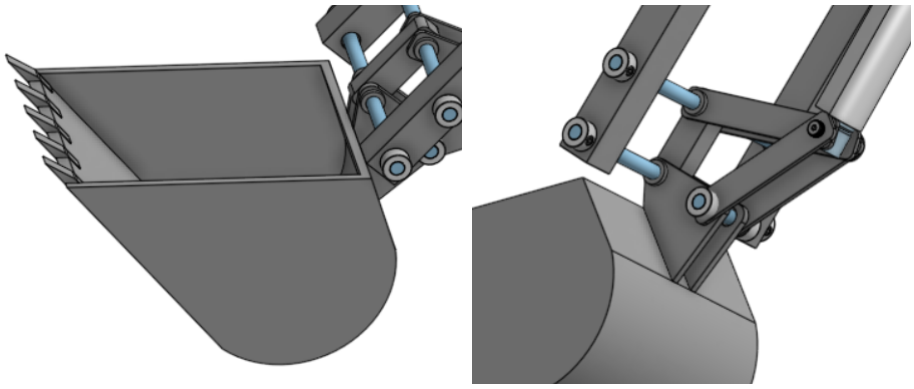

Figure 11: Excavator Scoop

3.1.3 Delivery System

Our delivery system is a simple container. It is designed to fill as much space as possible while fitting within the constraints of our digging system (Figure 12). It fits 13.6kg of material and is lifted using a sprocket and chain. We made it as simple as possible since it only has one function.

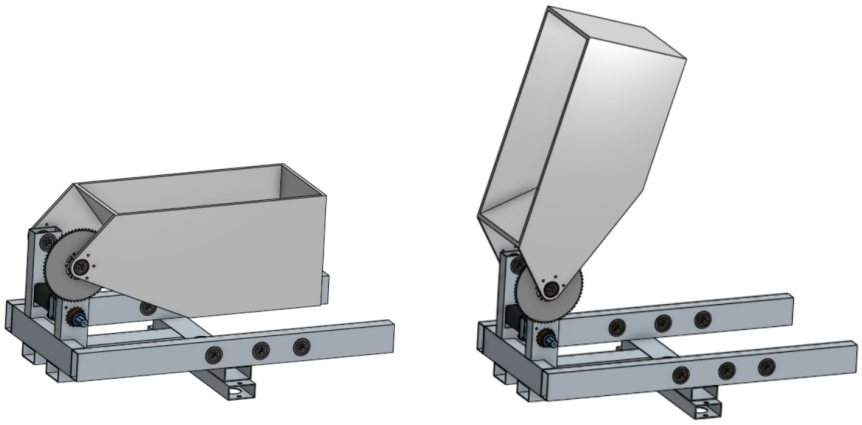

Figure 12: Dumping Bucket

3.2 Electrical

This section will present the various electrical components of the robot, starting with the actuators then moving to the sensor suite and concluding with the computer hardware needed to control and communicate with the robot.

3.2.1 Actuation

Based on the mechanical design, there are 5 components in the robot that need actuation. These are the dumping mechanism, the four-bar mechanism, the scoop control, the wheels, and the camera mount. The first is the dumping mechanism, controlled by a FRC Bag motor and a 300:1 gear ratio, split between a gear box and a additional chain drive reduction. The second two components together control the digging abilities of the robot. The four-bar mechanism is controlled by a FRC Bag motor and a 300:1 gear ratio, with a gear box and an additional spur gear reduction. The scoop is controlled by a PA-14P linear actuator to allow for high-precision motions under a large amount of force. As the calculations in the analysis section will show, the linear actuator must exert over 100 lbs of force without back-slipping.

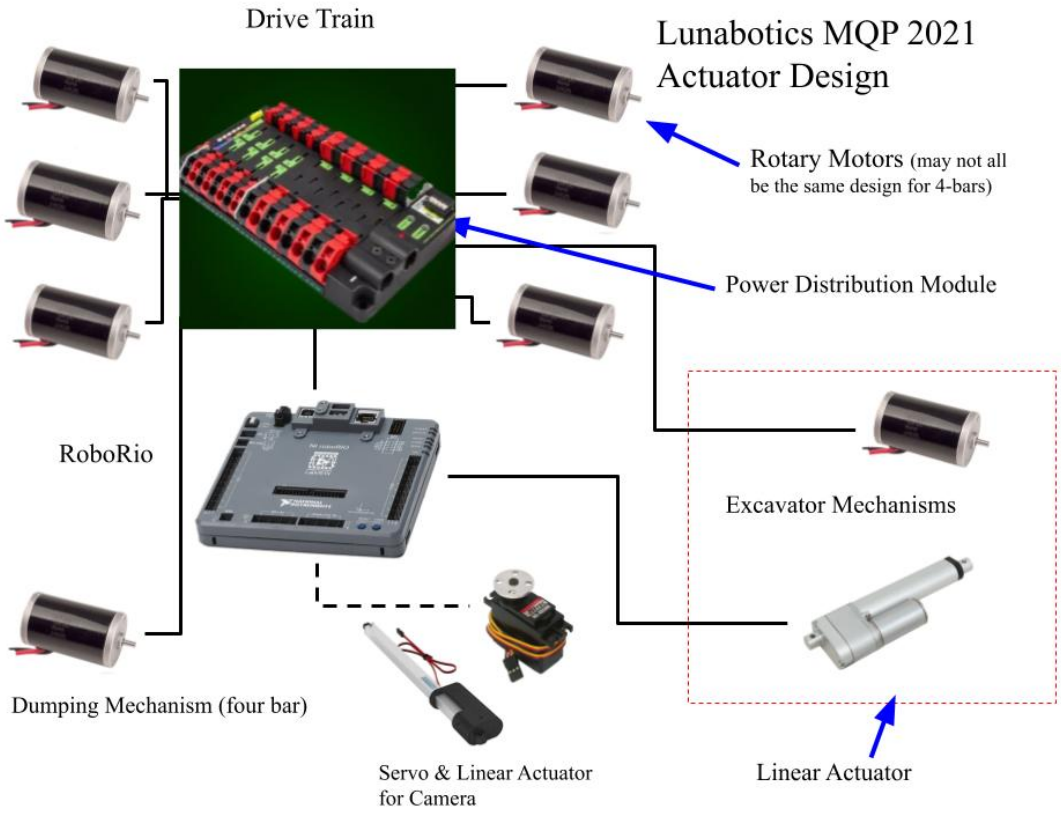

Figure 13: Concept Diagram of Actuators and Connections

The fourth component is the wheels. They also use FRC Bag motors, with a gear ratio of 87:1. The four rear wheels utilize a cantilevered design, with the motor and gear box in line with the drive shaft. Due to a physical conflict with the excavator, the front two wheels instead utilize a 90deg bevel gear and chain drive to allow for the placement of the motor above the wheel. The final component is the camera. Because of the challenges in mounting a camera with a clear view that does not interfere with moving components of the robot, the camera itself is extendable. It is mounted to a P-16 linear actuator whose base is mounted

to a simple servo. This setup allows the camera to rise above the other components in the robot for navigating, and obtain a full view of the field when the robot is going through the initialization and orientation process. It also keeps the robot within the height requirement and moves the camera out of the way when excavation is occurring.

3.2.2 Sensing

The sensing capabilities of the robot can be broken up by three distinct functions. These are dumping, excavation, and navigation. The dumping mechanism always starts in the same position, is driven by a DC motor and it is not a high precision subsystem. Therefore, a SRX rotary encoder was included between the gearbox and the additional chain reduction, allowing for a more compact physical design and high resolution sensing. A load cell with an analog amplifier was also included to allow remote/autonomous sensing of the amount of material in the collection bin.

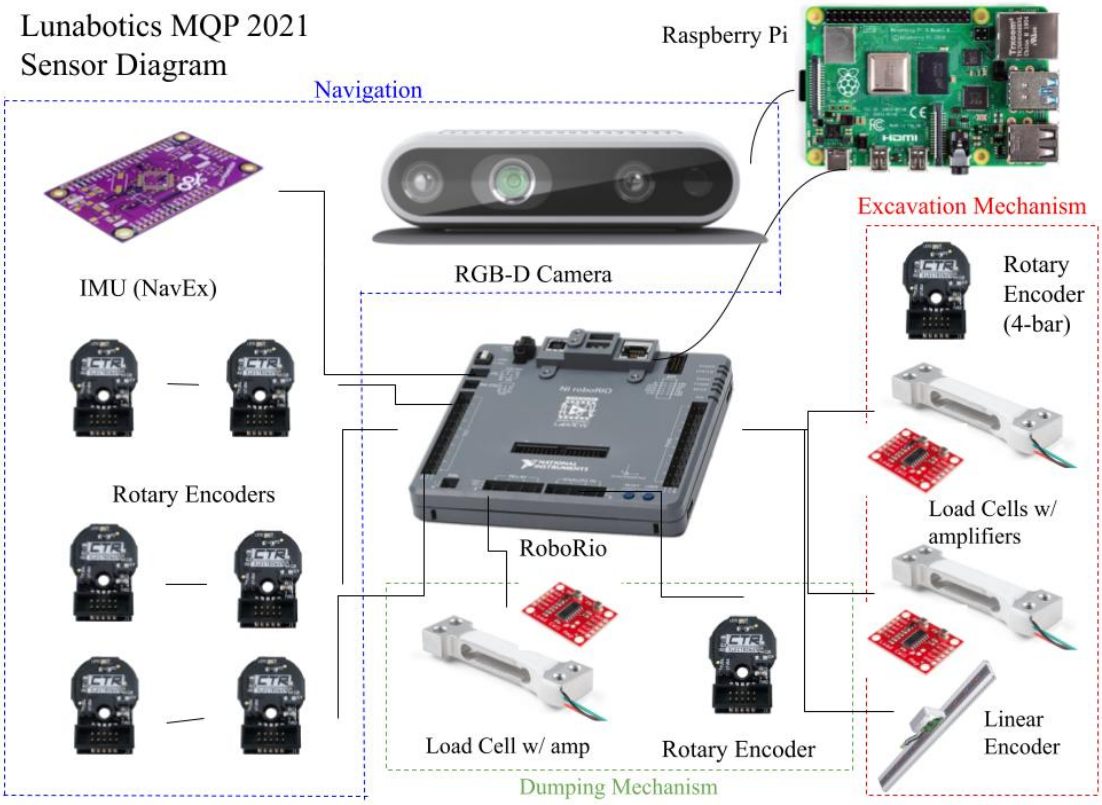

Figure 14: Concept Diagram of Sensors and Connections

The excavation subsystem contains two actuated parts, and both need to provide feedback in order for the robot to have an accurate concept of the physical position of its components. This was achieved by including an SRX rotary encoder between the gear box and spur gears

on the four-bar drive train, much like the dumping mechanism. The need for feedback for the scoop mechanism led to the choice of the PA-14P linear actuator, which provides feedback from an internal potentiometer. To better understand the state of the robots excavation, the original design incorporated two load cells into the coupler link of the four-bar mechanism. These would theoretically provide more detailed information on the status of digging, allowing the robot to sense if the scoop was stuck and how much material the scoop had collected.

The final sensing capability of the robot is navigation. The Intel RealSense D435I RGB-D camera does a large amount of the work on this front. It has a software-based IMU, object recognition, simultaneous localization and mapping (SLAM), and other functionalities that can be implemented through the software IDE. In addition, there are a couple other sensors that contribute to navigation and position information. The first is a hardware IMU, the navX chip (Invensense MPU-9250) often used by FRC teams. This allows for the IMUs to be cross-checked, which is particularly beneficial because the camera can rotate independently of the robot. The final sensor is actually six, the SRX rotary encoders placed on each wheel of the robot. These allow for the robot's distance travelled to be tracked, and placing encoders on all six wheels will allow multiple data points to be collected. Once outliers in the data are removed, the final set can be averaged. This will lead to more accuracy as the robot travels over rough and slippery terrain. Overall, this suite of sensors will allow for accurate recording of the position of the robot's components, and the material collected, and a good estimate of its location and surrounding in the competition field.

3.2.3 Network

The robot also comes equipped with a router as part of both the challenge requirements and as a crucial tool for communicating with the robot during testing and deployment. The challenge requires that the robot, and the control software, are connected to a local network to simulate conditions to that of a real life mission where the operator can only interact with the rover by network communication. The router also provides a more practical benefit, it allows the control software to interact with the robot remotely to simulate the competition environment where data will be communicated via a wireless network connection. With this design choice, the control systems in the robot software must be able to perform their tasks while communicating over a network.

3.3 Software

The software design of the robot can be broken into three major components: the hardware abstraction layer, the state machine and control layer, and the navigation and object detection layer. This composition provides the robot with control over its hardware, decision making, and sensory data acquisition in an effort to automate the decision making process.

3.3.1 Hardware Abstraction Layer

As a stretch goal, the team also incorporated plans to implement a Hardware Abstraction Layer (HAL) for interfacing with the embedded computers on-board the robot. While a fairly large task in an of itself, providing a HAL would be hugely beneficial to current and future teams working on the project as hardware issues become easier to isolate and avoid. It also allows other members working on the software layer to avoid interfacing with the hardware at a low level.

The proposed HAL interface primarily encompasses the roboRIO, Raspberry PI v4, linear actuators, motors, encoders, load cells, navigation camera, and the router. With the low-level details abstracted away, members working on complex tasks such as navigation, object detection, and the state-machine can rely on the interface providing them reliable data and state updates. This also prevents the situation where future development has to work with the low-level code which has a higher chance of producing error-prone code.

roboRIO:
The HAL provides an API for accessing hardware features such as interrupts, I2C, SPI, PWM, Ethernet, USB, and GPIO.

Raspberry Pi V4 Model B:
The HAL provides access to useful hardware features present on the Raspberry Pi such as PWM, Ethernet, GPIO, and interrupts.

Linear Actuators:
The HAL provides access to functions that modify and read the state of linear actuators used on the robot.

Motors:
The HAL provides an interface to control and read the state of the on-board motors

Encoders:
The HAL provides an interface to calibrate and read encoders mounter on the robot

Load Cells:
The HAL provides an interface to calibrate and read measurements from the load cell on the bucket

D435i Realsense Camera:
The HAL provides an interface to operate the camera such as requesting new frames and gathering IMU data

Linksys Router:
The HAL provides an interface to establish network connections via the router and connect on-board devices together

3.3.2 State Machine

The state machine layer facilities the main decision making of the robot in an effort to make autonomous decisions in terms of navigation, sensor or motor adjustment and reading, and error recovery. The state machine is one of the larger modules of code within the software portion of the project as it connects most of the smaller software tasks together. Using the Yakindu State Chart Generator, the state machine was designed to control all aspects of the robot in an autonomous fashion including initialization, routine state polling, error handling, navigation, and excavation. The state chart tool allows for code related to the state chart to be developed and tested independently of other modules such as the server, HAL, or navigation algorithm. Because of this, the state machine is also designed to handle different configurations for things like response times, and frequency of updates and request so that regardless of how other hardware and software are configured, the general flow is maintained. The breakdown of the state machine can be seen below and is primarily composed of these main modules: A high level overview of the state machine can be seen in Figure 15 on the next page.

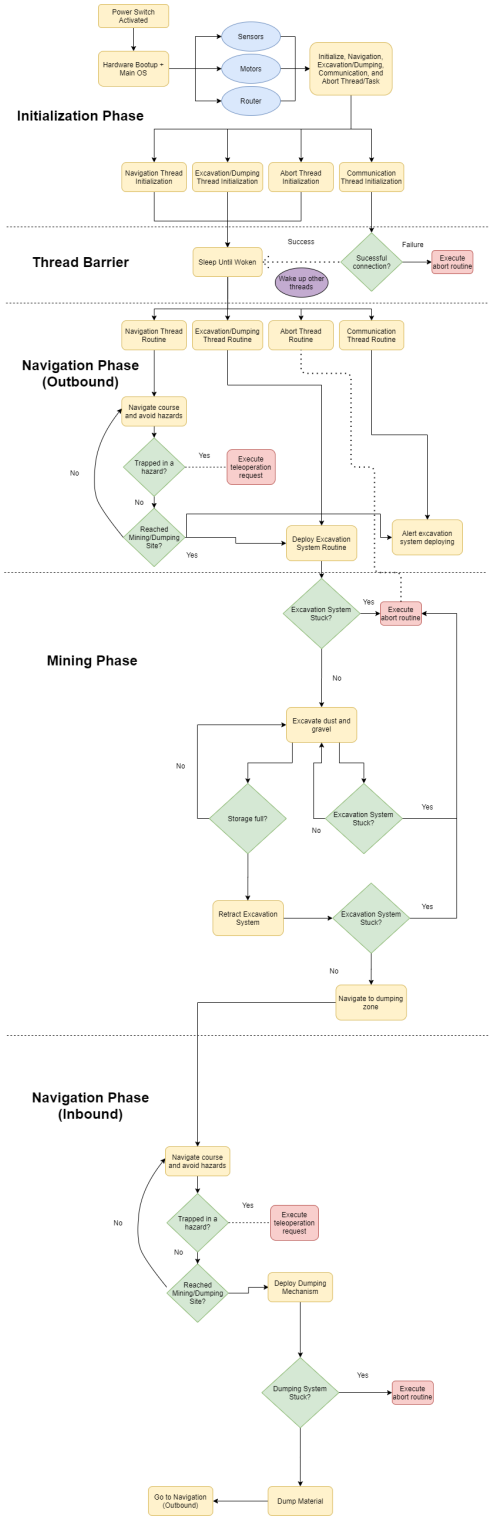

Figure 15: High Level Overview of Autonomy and State Machine

Initialization:

Within this layer, startup routines are executed as the on-board computers start the main processes. These include testing readings from sensors, motors, the server, the camera, and other service routines that help the robot boot up.

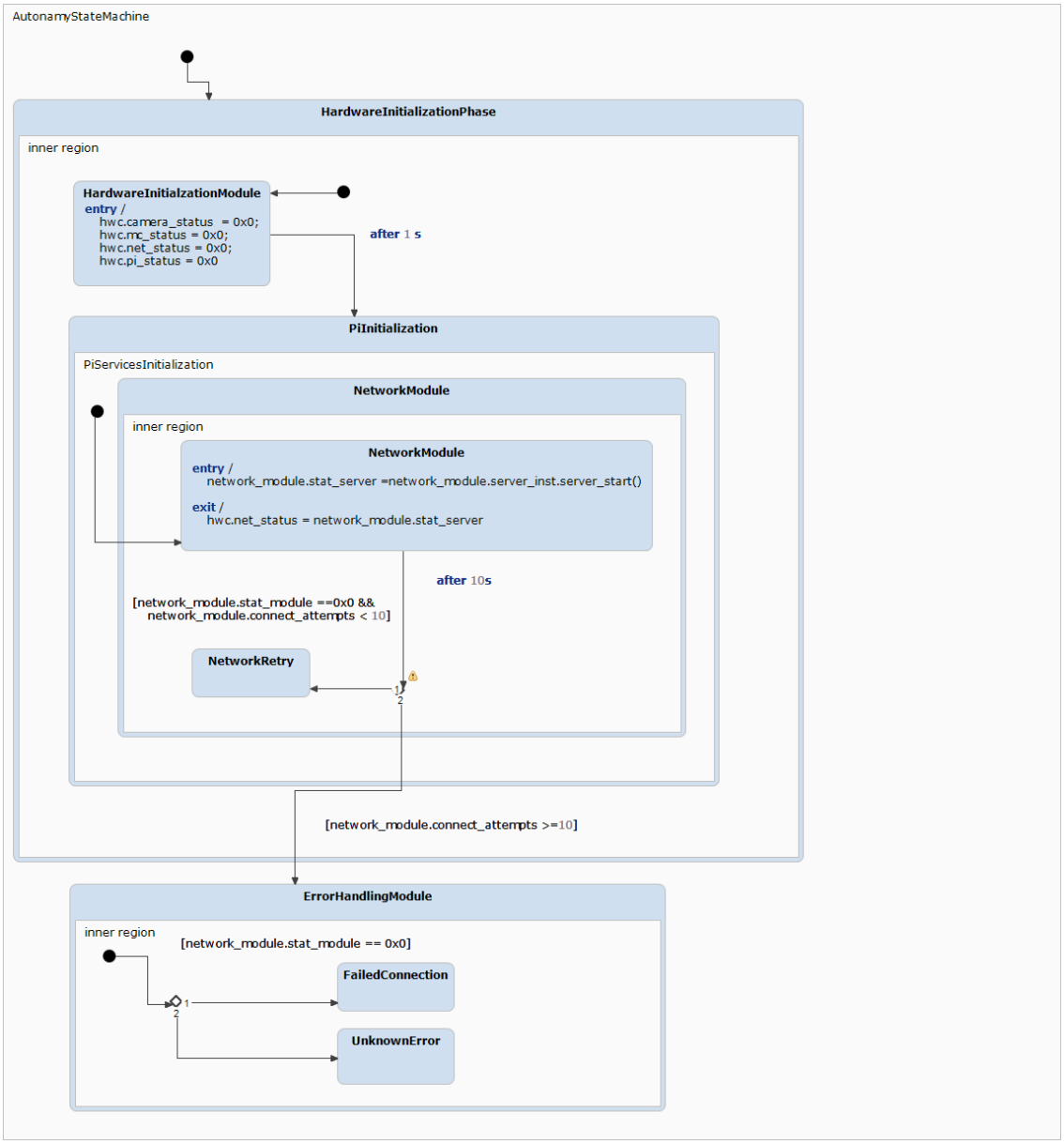

Figure 16: Overview of State Machine Layer: Initialization

Localization:

This layer describes the states and actions the robot goes through when attempting to localize its surroundings. Routines such as raising, rotating, and capturing from the Realsense camera are executed here. Additionally, a SLAM application is executed/interacted to continuously develop an internal map of environmental data used in navigating.

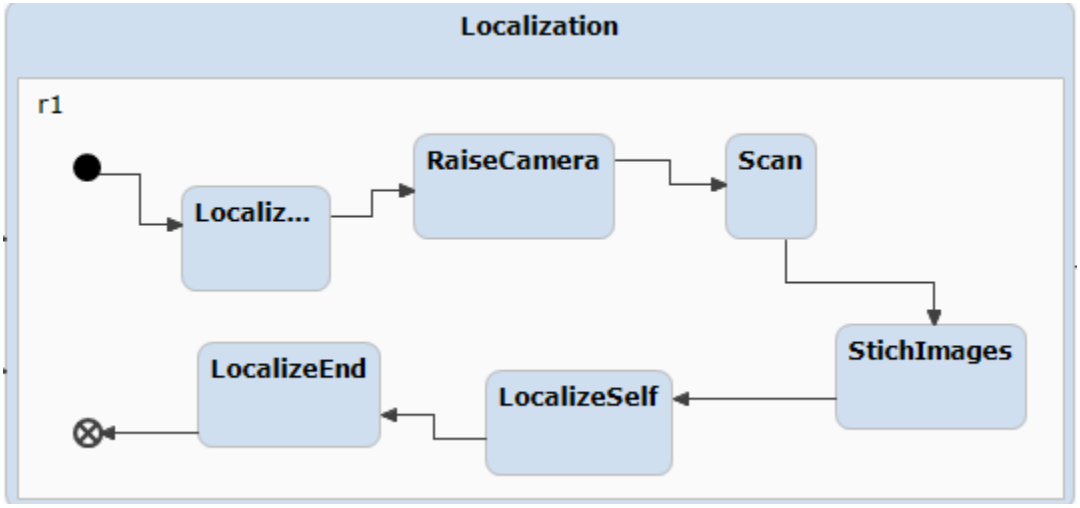

Figure 17: Overview of State Machine Layer: Localization

Navigation:

This layer handles moving through the immediate environment, utilizing the on-board sensors and navigation algorithm to determine a safe and efficient path to a target: either the mining zone or dumping zone. Tasks such as obstacle avoidance are also handled in this layer.

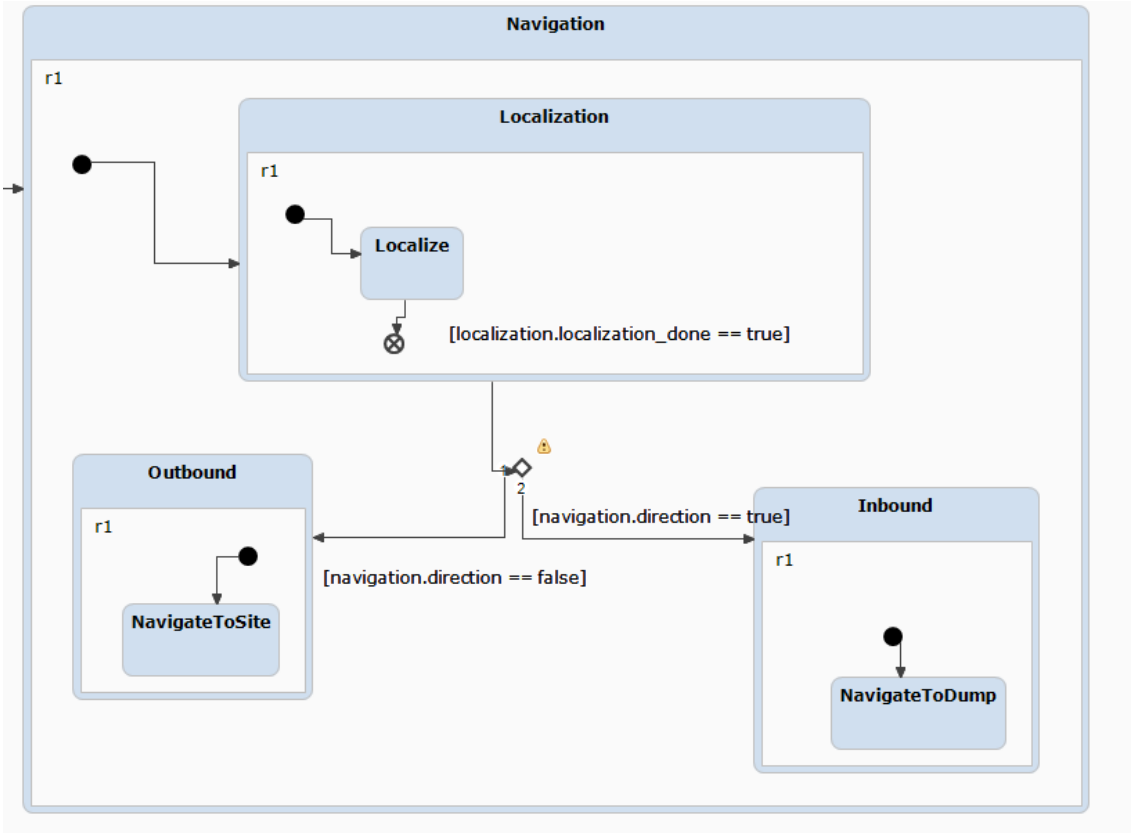

Figure 18: Overview of State Machine Layer: Navigation

Excavation:

This layer handles the excavation of Regolith material when the robot reaches the mining zone. This portion of the state machine controls the linear actuators to collect material in front of the robot. To determine the state of the excavation process, the state machine will measure load cell reading, the current draw from the both the BAG motor and linear actuators, and the encoder feedback. This will provide enough information to determine whether it is successfully collecting material, depositing it in its bucket, and if it is getting stuck on something.

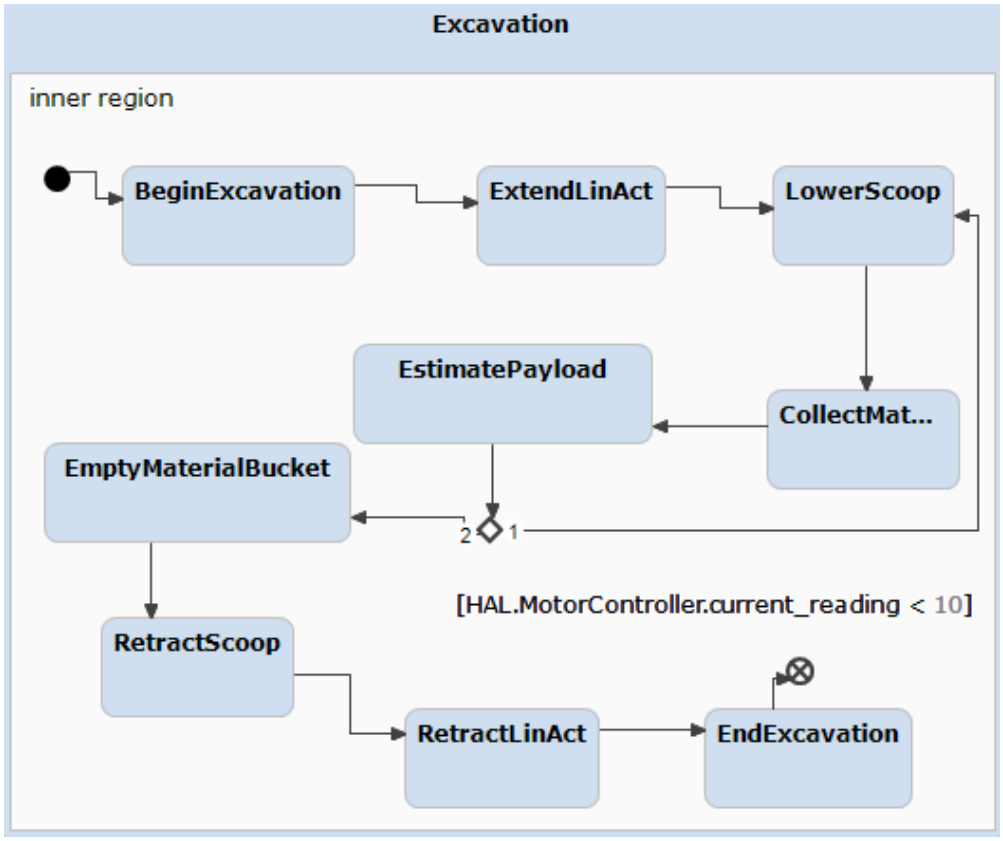

Figure 19: Overview of State Machine Layer: Excavation

Dumping:

This layer handles dumping material into the target bucket when the robot has collected enough material. Much like the excavation layer, a variety of sensor readings are monitored to determine the state of the dumping process as the linear actuator(s) empty the material into the target bucket.

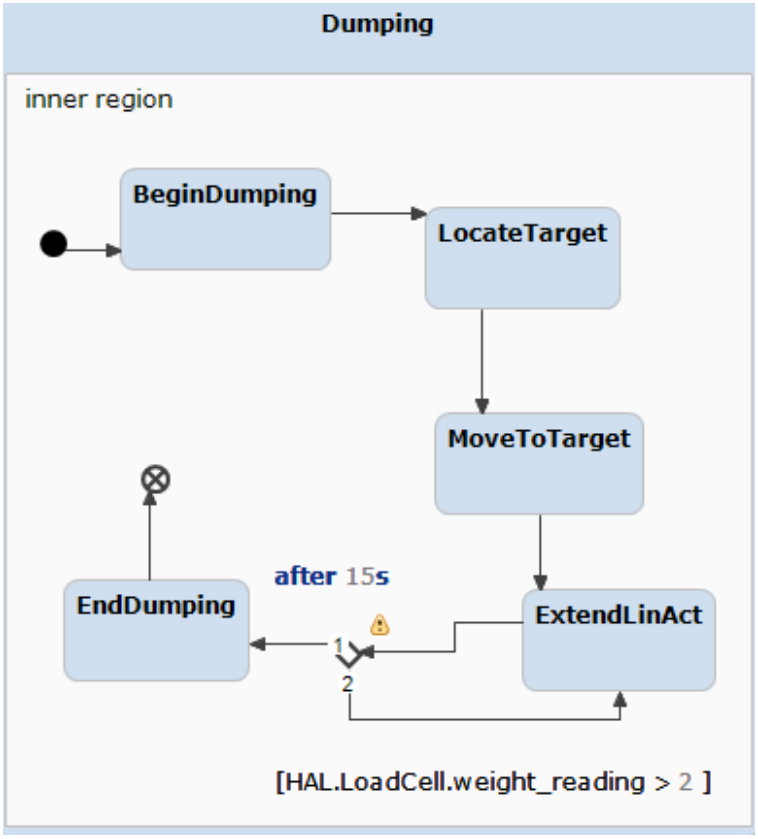

Figure 20: Overview of State Machine Layer: Dumping

Network and Communications:

This layer handles all network communications the robot receives and transmits. This layer operates asynchronously as telemetry data is periodically sent back to the control computer. It also offers teleoperation control, allowing the control computer to directly operate the rover.

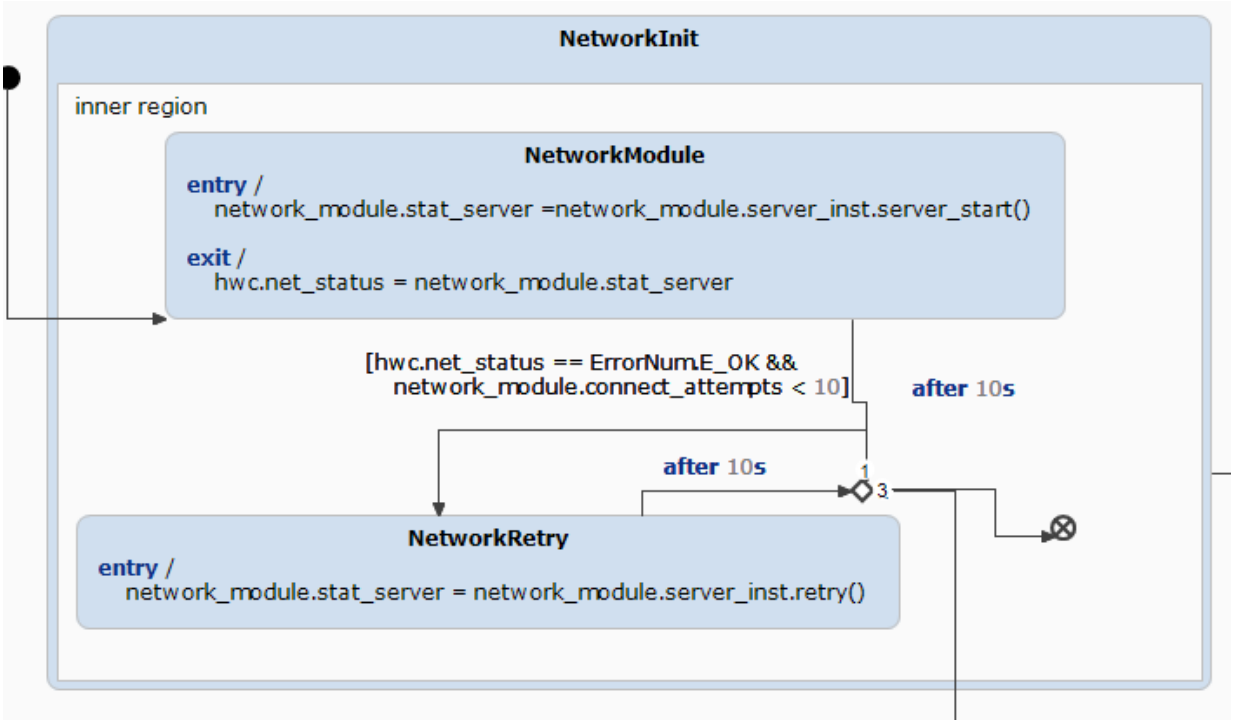

Figure 21: Overview of State Machine Layer: Network and Communications

Error Handling

This layer handles all major error handling and recovery for the robot. While each of the other layers performs some form of error handling and recovery, this layer's primary responsibility is to restore the robot to the last successful state in more dire situations and communicate all useful data to the control computer. Effectively, this layer attempts to find where the problem is, determine whether it can recover, and execute a plan of action such as shutdown or restoration.

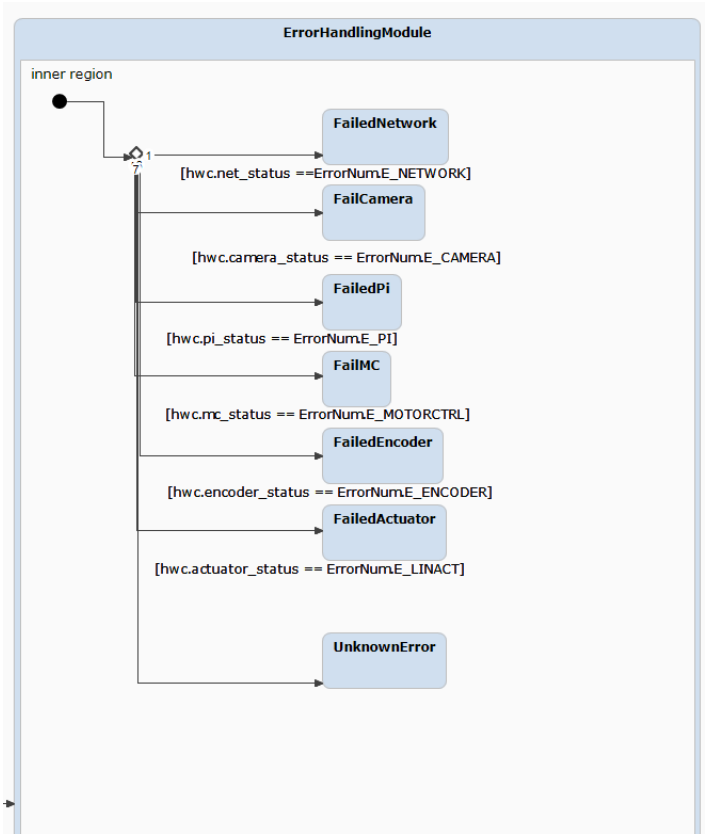

Figure 22: Overview of State Machine Layer: Error Handling and Recovery

In general, the state machine's purpose is to handle the decision making the robot will need to make in order to facilitate autonomy for at least a single cycle. The state machine is designed in such a way that is its independent of the actual implementation of the specific routines run by the robot, such as extending the linear actuator. This was done so that development of either system could progress regardless if access to one or the other was impractical (as was the case at the beginning of C-term 2021).

3.3.3 Navigation

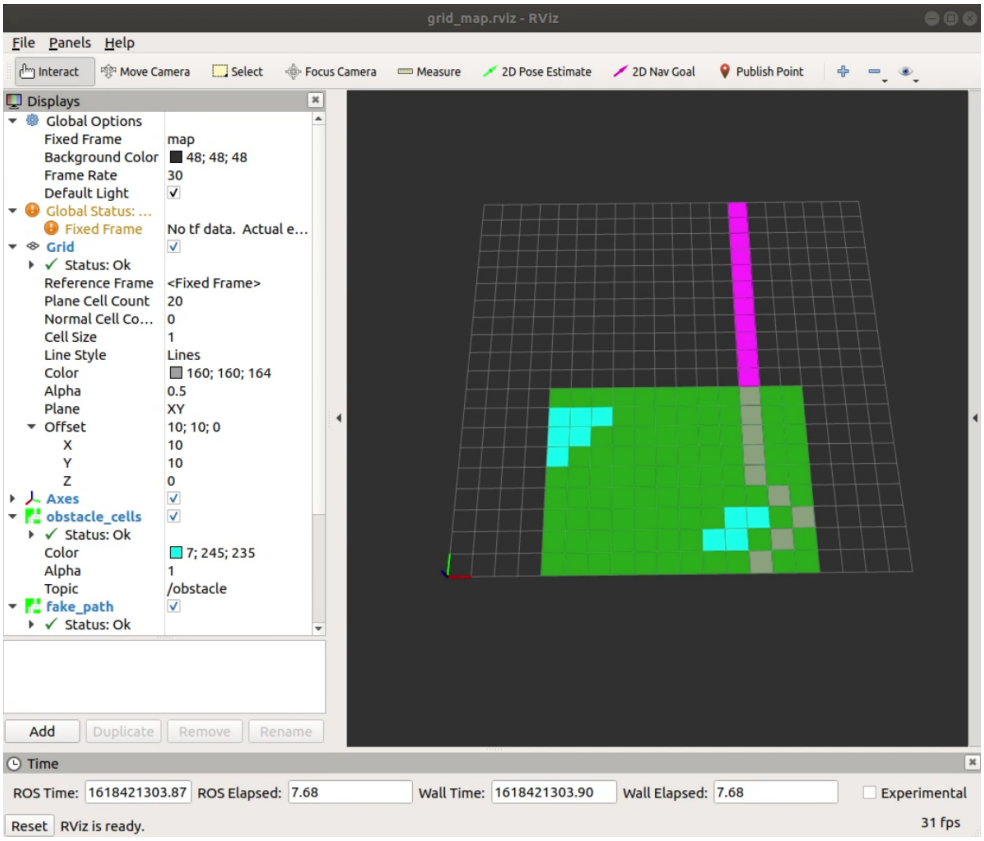

Figure 23: Exploration and Navigation Program in Progress

The navigation layer handles incoming sensory data and transforms it into both machine-readable and human-readable map data to be used in finding the current optimal route to a specific target. This layer also handles recognizing hazardous obstacles such as craters and rocks and identifies them for the navigation system to avoid. SLAM is performed by the code package that comes with the real-sense camera. The most updated terrain data is then sent through a ROS bag file to custom ROS nodes. The data consists of information about the terrain, and this data is categorized by grid cell and displayed in RVIZ as an obstacle, open space, or unknown. This data is also used to compute the best navigation path using the Astar algorithm, which is also displayed in RVIZ. This results in a human readable map that can be integrated with an interactive display in the future.

4 Methodology and Testing

4.1 Excavator Construction

In order to validate our design, we chose to build the excavator subsystem. This is because it would be easier to test and get physical results of its digging potential. Using the original CAD, the team designed a wooden chassis that attaches to the four-bar in figure 24. This allowed us to examine the forces on the entire system while digging.

Figure 24: CAD of excavator with wooden base

The team chose welding as the preferred manufacturing method for fastening the aluminum tubing. The original design was to be made from 1/16 inch tubing; however, this was changed to 1/8 inch to make the welding easier(figure 25). The aluminum plates used for the excavator bucket were outsourced to SendCutSend, a company that custom manufactures various types of metal. The curved back wall of the bucket was made from a 1/8 inch plate of aluminum that was shaped to the desired curve. The five aluminum teeth were taken from parts from a previous year's robot. For the test base, the team chose to use wood 2x4s and screws for ease of manufacturing.

After all the welds were finished, the linear actuator arrived. To the team's surprise, it was 3 cm too long. This affected our attack angle of the scoop and needed to be corrected.

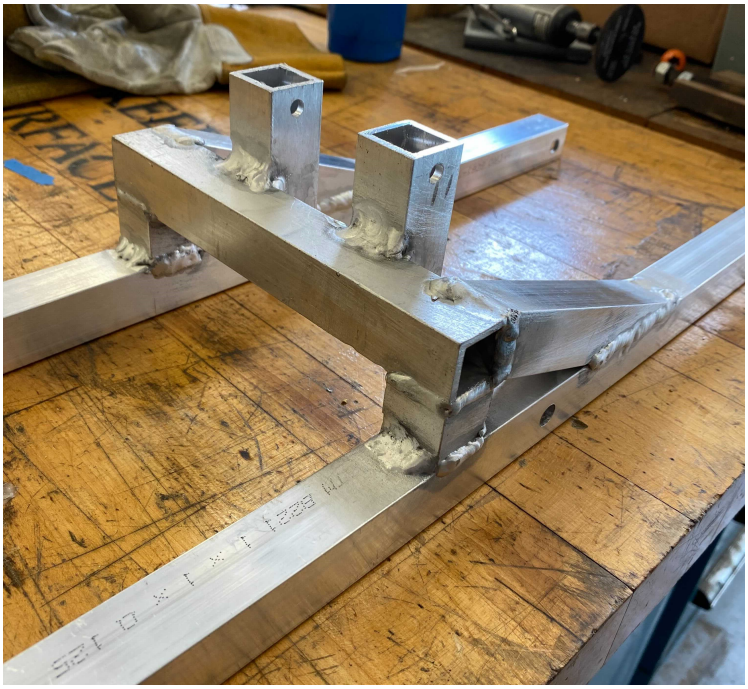

Figure 25: Welded support bar on main links

After testing different modifications of the design, the team found the best solution was to extend the attachment of the actuator backwards. This was completed by welding on two additional 1 inch tubing pieces with holes drilled slightly offset from the center (Figure 27). This realigned the built excavator with the range of motion in the CAD model.

Figure 26: Welding on bucket (before teeth)

Figure 27: Extended back attachment on linear actuator

The four-bar was assembled with plastic bearings, bolts, steel axles aluminum versa-hubs, and shaft collars. Once the four-bar was completed, it was attached to the wooden base. The crank link was connected to the chassis by an aluminum hex axle and the follower link was connected by a steel axle. The motor was mounted to the wooden base, and the final excavator is shown in Figure 28.

Figure 28: Built Excavator

4.2 Excavator Testing

Testing the excavator is imperative to ensure the team's design could fulfill the competition requirements. Although the competition was cancelled, knowing how well the excavator can perform will also be beneficial for future WPI teams that may consider using our design. To be sure that the design was effective, the students decided to test the excavator in two different ways. The first was to test how well the excavator performed digging and dumping. The second was to determine the forces acting on the frame in the horizontal direction during the digging process, standing in for how much friction the wheels of the rover would need with the ground.

Determining how well the system could excavate sand was the most important metric to measure. Sand was used as a stand-in for BP-1, the lunar simulant used in competition. Understanding the excavator's digging capabilities was necessary to know if the four bar design is feasible for the competition. To determine how well the excavator could perform, the students took three measurements. The first was the quantity of sand excavated in each scoop. Each scoop of sand was dumped into a box and then weighed on a scale to determine how much sand was excavated. The second measurement was to determine how many scoops it takes to get to the gravel at a depth of 30cm. The third was how long it takes to complete one full cycle. The team felt that these three measurements would determine whether the four bar excavator design was adequate to mine enough gravel to be awarded a good score in the competition.

The second major area of testing was force analysis. The team believed it was important to know the horizontal force acting at the base of the excavator. This force is important to know because it will tend to drag the robot forward during the digging process. If the robot moves forward then the excavator will not dig in the same spot each time. To determine this force, the base of the excavator was placed on two drawer slides so that friction could be considered negligible. A spring scale was then attached to the back of the base at one end and secured to the crossbar of a table on the other end. The spring scale was read during the digging process to determine the max horizontal force acting on the base of the excavator.

4.3 Software Build

The final deliverables for the software portion of this MQP involved creating a simple Hardware Abstraction Layer (HAL) to interface with the various sensors and motors, a state machine to control various routines the robot performs autonomously, and the path finding and navigation program.

4.3.1 HAL Implementation

As outlined in the design section, the HAL is one of the objectives the software team worked on as part of the final deliverable. This was done to streamline current and future efforts in regards to developing the control systems that interface with embedded hardware components. Additionally, because of the primary mechanical deliverable of the excavation

system, software development shifted towards tools and programs that would best support the mechanical and electrical requirements in order to create a working prototype. In addition to writing new interfaces, we also aim to include the use of popular 3rd party libraries such as WPILib and CTRE Talon SRX libraries. These provide useful abstractions and definitions for existing hardware that helped improve the reliability and effectiveness of our code.

The code for the HAL portion of the excavation programs can be broken down into a number of smaller modules:

Initialization:
The initialization module provides an interface to quickly set up and initialize specific sensors and devices for use in the rest of the program and robot.

Sensor and Device Reading/Writing:
The HAL also provides simple device access to read and write data for use in robot code.

Teleoperation:
In addition to an autonomous interface, the HAL also incorporated a teleoperation system to manually control robot systems for testing and debugging.

Debugging and Logging:
A debugging and logging system was also developed for the HAL.It provides both run-time and post-run support. An interface to read and write to the logging system was provided to use during testing and code development.

4.3.2 Initialization

Like most embedded system projects, a large portion of the program involves "boiler-plate" code to correctly set up, manage, and control the various devices connected to the roboRIO. For the four-bar and scoop system, devices such as magnetic encoders, motor controllers, power distribution panel, and the main linear actuator and BAG motor all required specific code to properly initialize their respective hardware for use later on.

The most important devices that are initialized are the CTRE Talon SRX motor controllers that provide easy access to control systems for the BAG motor and linear actuator that move the excavation system. An example of some of the initialization done for the Talon's can be seen below in Figure 29.

As seen in the code snippet, two instances of motor controller objects are initialized with specific settings such as what type of feedback the device provides, how its absolute position should be delivered (positive/negative), and other settings that allow the device to be properly configured for use. These two motor controller instances are then used throughout the rest of the program to control motor outputs for both the linear actuator and the BAG motor. Additionally, these controllers are configured to communicate via the Controller Area Network (CAN) protocol for fast and reliable messaging.

```
void Robot::InitializeTalonLinearActuator() {
  SRX_FOURBAR.ConfigFactoryDefault();
  SRX_LINACT.ConfigFactoryDefault();
  SRX_FOURBAR.ConfigSelectedFeedbackSensor(FeedbackDevice::CTRE_MagEncoder_Relative,0,10);
  SRX_LINACT.ConfigSelectedFeedbackSensor(FeedbackDevice::CTRE_MagEncoder_Relative,0,10);
  /**
      * Configure Talon SRX Output and Sesnor direction accordingly
      * Invert Motor to have green LEDs when driving Talon Forward / Requesting Postiive Output
      * Phase sensor to have positive increment when driving Talon Forward (Green LED)
      */
  SRX_FOURBAR.SetSensorPhase(false);
  SRX_FOURBAR.SetInverted(false);
  SRX_LINACT.SetSensorPhase(false);
  SRX_LINACT.SetInverted(false);

  /* Set relevant frame periods to be at least as fast as periodic rate */
  SRX_FOURBAR.SetStatusFramePeriod(StatusFrameEnhanced::Status_13_Base_PIDF0, 10, 10);
  SRX_FOURBAR.SetStatusFramePeriod(StatusFrameEnhanced::Status_10_MotionMagic, 10, 10);
  SRX_LINACT.SetStatusFramePeriod(StatusFrameEnhanced::Status_13_Base_PIDF0,10,10);
  SRX_LINACT.SetStatusFramePeriod(StatusFrame::Status_10_MotionMagic_,10,10);
```

Figure 29: Example of initialization code

4.3.3 Sensor and Device Reading/Writing

To facilitate actually interacting with the hardware, methods to read and write data to specific devices were created. This happened for any device that required communication past the initialization phase such as motor controllers, analog input, joystick input, and the power distribution panel. Our interface provides a non-blocking method to read and write data to these devices so as to not delay the overall control loop when reading and writing.

Most of the reading and writing values is handled via interfacing with the WPILib and CTRE libraries to communicate with the roboRIO and the Talon SRX's. To ensure the most up to date information, the state machine reads the current, motor output, and encoder feedback from the motor controllers frequently. These motors are controlled via setting either a proportional motor output or absolute position, and the motor controller handles regulating the correct voltage output to achieve the desired position. Device feedback is included in this module as two different feedback methods were used to adjust and compute new output values to send to the motor controllers.

The first feedback method was generated from the linear actuator controlling actuating the scoop, which sent back a analog value determined by a 10kΩ potentiometer detailing the number of complete turns the actuator had completed at the current time. The derived value after ADC conversion is seen below when the ADC is configured as 12-bit ADC with 4-bit oversampling. The total range of turn readings is [0,10], where 0 means completely retracted and 10 signifies complete extension. As with most ADC conversion, the output of the ADC is based on the V_{ref},V_{in}, and the bit width of the ADC. The roboRIO analog inputs support 12-bit ADC conversions which is more than enough to properly represent the resolution of the sensor output.

$$ADC_{12\ Bits} = \frac{V_{Linear\ Actuator} * 4096}{5.0}$$

After the ADC conversion to its digital equivalent, we then scale the output to a smaller range to identify how many complete turns the linear actuator had completed, as shown below.

$$r_{max} = 4096, r_{min} = 0, t_{max} = 10, t_{min} = 0$$

$$Number\ of\ Turns = \frac{ADC_{12\ Bits} - r_{min}}{r_{max} - r_{min}} * (t_{max} - t_{min}) + t_{min}$$

This is just an example of the some of the simplification the robot software provides as operations like the one above are simply abstracted to a simple function call to read a specified analog port. This sensor reading can now be used in both the general control system and the logging or autonomous portions of the code without worrying how to actually read the sensor registers.

4.3.4 Teleoperation

To provide manual control over the robot's subsystems, a teleoperation mode was implemented for both testing and manual recovery. This mode was developed early relative to the autonomous module as it was necessary to test and familiarize the team with the control system hardware. Using a Logitech gamepad as the controller, a user could operate a variety of hardware features such as control over both the linear actuator and BAG motor via interacting with the two motor controllers, print debug information of the current state of the robot, and re-initialize sensors to reset the system. Two control systems were implemented to allow the controller to operate the motor controllers: motion magic and percent output. Motion magic provides an absolute position based on the attached magnetic encoder, while percent output merely regulates the output voltage based on a passed in value from [-1.0,1.0].

4.3.5 Debugging and Logging

The debugging and logging system provides access to recording and monitoring the robot's state via a runtime logging interface that utilizes powerful features of the WPILib interface. With the current build of the logging system, real-time updates to the `frc::SmartDashboard` are supplied from both periodic routines and during important events such as state transitions, certain teleoperations, and critical warnings like current draw. With this, the team was able to easily monitor and debug both teleoperation and autonomous performance during testing and development to allow for fast and accurate changes to be made. It also provides a clear picture of how the robot is performing in terms of how accurate and precise the control system is in regulating the usage of the motor controllers. An image of the debugging interface can be seen below in Figure 30 where a list of useful statistics are updated in real-time.

In addition to the real-time status updates, the autonomous system also leverages a state-transition based logging system that records certain metrics when the state machine

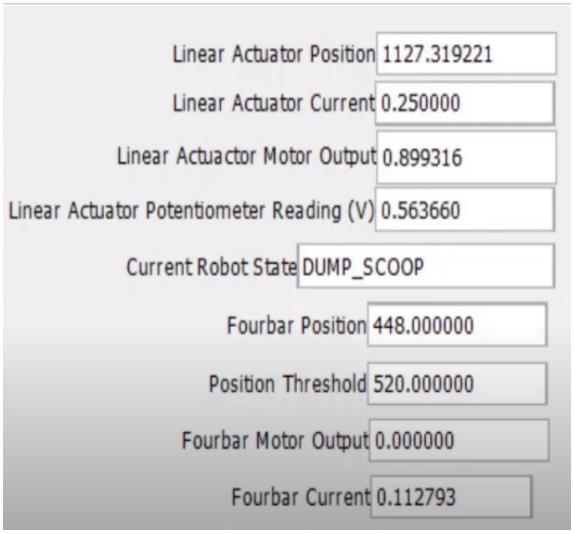

Figure 30: Smart Dashboard Interface

changes from one state to another. Important metrics such as time in state, average current draw, feedback error, and start and stop voltage can be recorded. The implementation of this system utilizes a similar mechanism to popular debugger programs such as GNU `GDB` with a stack data structure to store a chain of state transitions. This inherently records the timeline of state transitions, allowing for event-by-event inspection of each state transition. This system can be easily modified to accommodate more information and can be inspected both using the WPILib debugging interface and post-run updates to a standard output. An example of a single autonomous cycle state transition record can be seen in figure 31.

```
State[DUMP_SCOOP] Time[12] % Error[0%] Avg Current[0.140015 A] Min/Max Current[.2494A,.5A]
State[DIG_RETRACT_FOURBAR] Time[8] % Error[-18.2979%] Avg Current[0.139771 A] Min/Max Current[.1102,.755A]
State[DIG_EXTEND_SCOOP] Time[13] % Error[0.0540541%] Avg Current[0.101196 A] Min/Max Current[.2495A,-.626A]
State[DIG_EXTEND_FOURBAR] Time[9] % Error[-12.3420%] Avg Current[0.121216 A] Min/Max Current[.1121A,.702A]
```

Figure 31: Log informationfor single autonomous cycle

With this info, identifying key metrics such as time spent in a certain state or how accurate our control system proved to be easier.

4.4 State Machine

Another key component that the software team built was the state machine to control the four-bar and scoop during the digging and dumping process. As mentioned earlier, the focus of the state machine was narrowed to sections related to the digging and dumping process for this year. A working iteration of the state machine provided autonomous digging and dumping data for the team to analyze and report. The state machine is comprised of three

main components: data acquisition, decision making, and transitioning. Each autonomous iteration, the robot first gathers data about the current state it is in, such as the current state, current draw, encoder feedback, etc.; then, based on this information the robot requests the appropriate threshold values to determine whether a state transition is necessary. After this, the robot computes its new motor controller output values based on the desired next state and then signals to the motor controllers to output the desired value.

These states were represented by an `enum`, and a variety of methods were provided to manipulate and utilize their values for use in the state machine. The most important states were `DIG_EXTEND_FOURBAR`, `DIG_EXTEND_SCOOP`, `DIG_RETRACT_FOURBAR`, `DUMP_SCOOP`, and finally `RESET`. With the states, the state machine was successfully able to capture the necessary phases of digging and dumping with the four-bar and scoop design. With the combination of these states and the sensor information, the state machine is able to follow a simple decision tree to choose the appropriate value based primarily around the following (in order of importance:

1. Linear actuator/BAG motor position
2. Current draw from actuator/BAG motor
3. Current cycle in the autonomous loop

This hierarchy allows the state machine to identify important outlier values such as when a device's torque output increases. This signals to the state machine that it can transition to the next digging phase early to prevent potential damage. The state machine is able to isolate these events such that unless the outlier becomes sustained, the state machine will continue with its original state transition. This is useful when determining whether the four-bar has reached its digging spot and thus temporarily ignoring the desired target position is fine as attempting to reach said position would cause the motor to potentially burn out by attempting to push into the ground. In order for the state machine to make this important decision, a look-up table is implemented to store predefined sensor value. These values are then compared to the current state of the robot and act as thresholds. This meant that depending on which phase of the digging operation the robot was, the state machine and control system will dynamically alter the appropriate thresholds to accurately judge whether a given sensor reading was within valid range.

4.5 Navigation

Although navigation capabilities were not integrated with the test excavator, a great deal of development and programming was put into navigation capabilities. As outlined in the design section, the original goal for the robot was fully autonomous navigation. To this end, the team experimented with the Intel RealSense D435I RGB-D camera to map the robot's surroundings and ROS to run the Astar algorithm and handle the complex tasks required to navigate.

The Intel camera comes with the robust Realsense API that allows it to generate point clouds and map its surroundings. Depth data is gathered from the IR pattern returned by objects, and this information is what generates the 3D point cloud of what the camera sees. Through experimentation and testing, the image produced by the camera was refined. The API also enables Simultaneous Localization and Mapping (SLAM), which would integrate the point cloud data with camera's internal IMU to create a map that updates as the robot moves throughout the environment.

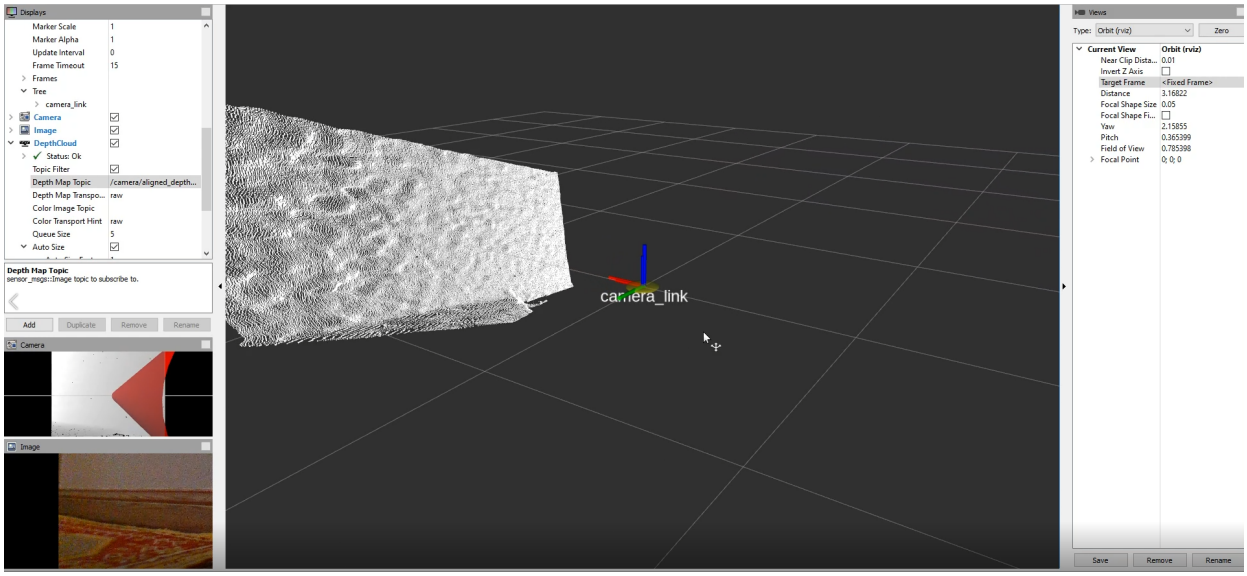

Figure 32: PointCloud with Depth Data Generated by Intel Camera

ROS was chosen as a platform for the higher level code because of its ability to freely share information among multiple devices. It was loaded onto the Raspberry Pi, the hub of the navigation system. The RGB-D camera output the SLAM information into a ROSbag file, which could be sent to the Pi and read in a ROS node. This data was then processed by the node to extract a map of the discovered area and obstacles. That map was displayed in RVIZ, a ROS graphical interface that can be integrated into later GUIs. The data was also sent to the Astar algorithm, which computed and returned a path to the other side of the field. This was also displayed with RVIZ.

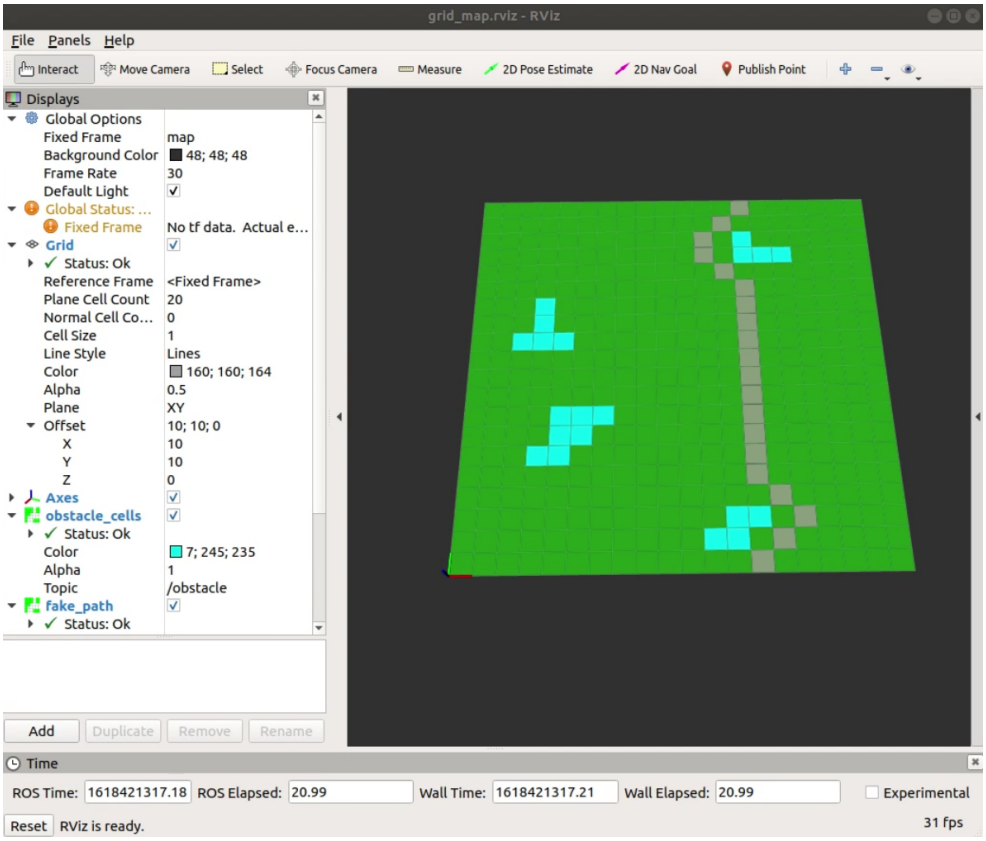

Figure 33: Test Exploration and Navigation Completed

This development was all done with the intention of integrating it within the larger state machine. Sensor data and information would move freely between the various ROS nodes, while the state machine would take the role of overseeing the process and deciding when to use the various capabilities of the robot. While a solid start to the navigation system, one big area that would still need to be solved is fusing the sensor and camera data to accurately predict and display the robot's position on the map, and translating the Astar path into commands for the robot to follow. However, these are challenges that we believe can be resolved within the code framework set out. We hope future teams can build on the foundation laid this year.

4.6 Software Testing

To test the developed software, digging routines involving both teleoperation and autonomous operation will be required to fully explore and evaluate the design. For teleoperation, simply controlling the four-bar and linear actuator with the previously mentioned controller to perform digging and dumping will be enough to demonstrate success. This test aims to validate whether the controller can correctly operate both devices, and allow the rest of the build, such as the four-bar and scoop design, to be tested in a semi-realistic environment.

The other area of testing will be examining the effectiveness of the autonomous capabilities of the digging system. Specifically, the state machine and control system will be tested for its ability to quickly and reliably collect both sand and gravel to estimate its effectiveness in a competition setting. By looking at how the state machine and control system handle the various stages of the autonomy cycle, our team can evaluate its ability to effectively dig and dump autonomously without relying on brute-force.

5 Analysis

5.1 Force Calculations

5.1.1 Digging Analysis

The maximum digging force F_{Dmax} occurs when the tip of the scoop is in contact with the ground and the robot starts tipping. In this case, the weight of the robot is supported by F_{Dmax} and the normal force at the rear wheels F_{nR}. Assuming that the robot does not carry any material collected in this situation and the mass of the robot is 40kg, the weight of the robot W_R will be 40kg * 9.81m/s^2 = 392.4N. The horizontal resistance experienced by the scoop during digging F_{Drag} is 10N as measured.

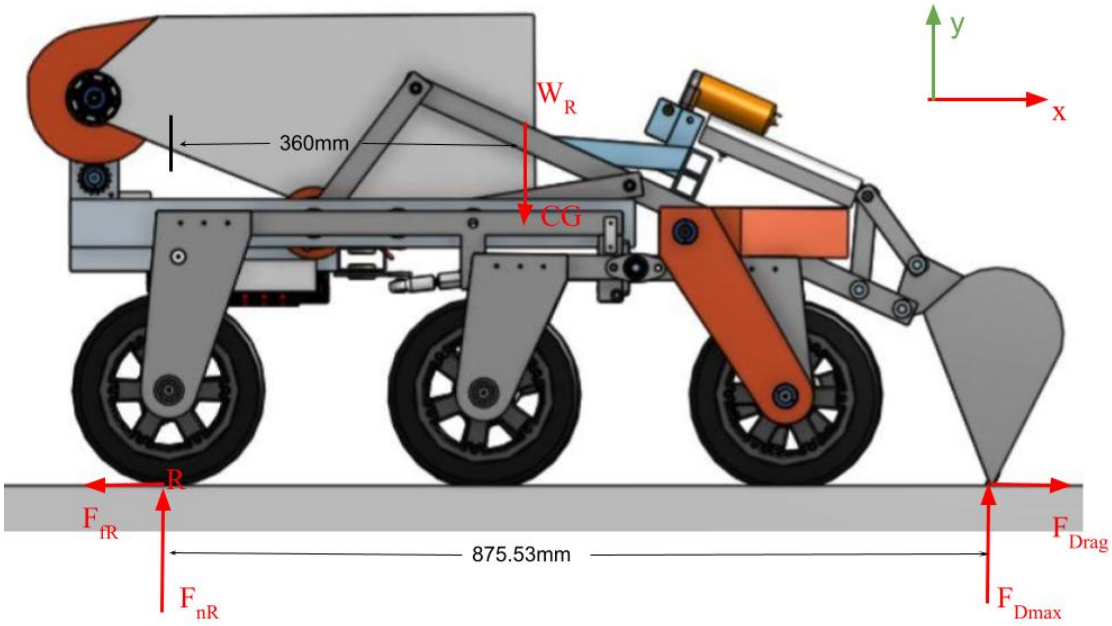

Figure 34: Free body diagram of the robot when the maximum digging force occurs

From the free body diagram of the robot when F_{Dmax} occurs, equations of equilibrium can be derived.

$$\sum F_x = F_{Drag} - F_{fR} = 0 \tag{1}$$

$$\sum F_y = F_{nR} + F_{Dmax} - W_R = 0 \tag{2}$$

$$\sum M_R = F_{Dmax} * 875.53mm - W_R * 360mm = 0 \tag{3}$$

By solving equation 1 to 3, the results are

unknown	value
F_{Dmax}	161.3N
F_{nR}	231.1N
F_{fR}	10N

After solving for forces acting on the digging scoop F_{Dmax} and F_{Drag}, the entire digging mechanism and each member within it need to be analyzed to determine the torque and forces required to actuate the mechanism as well as the loads at the joints.

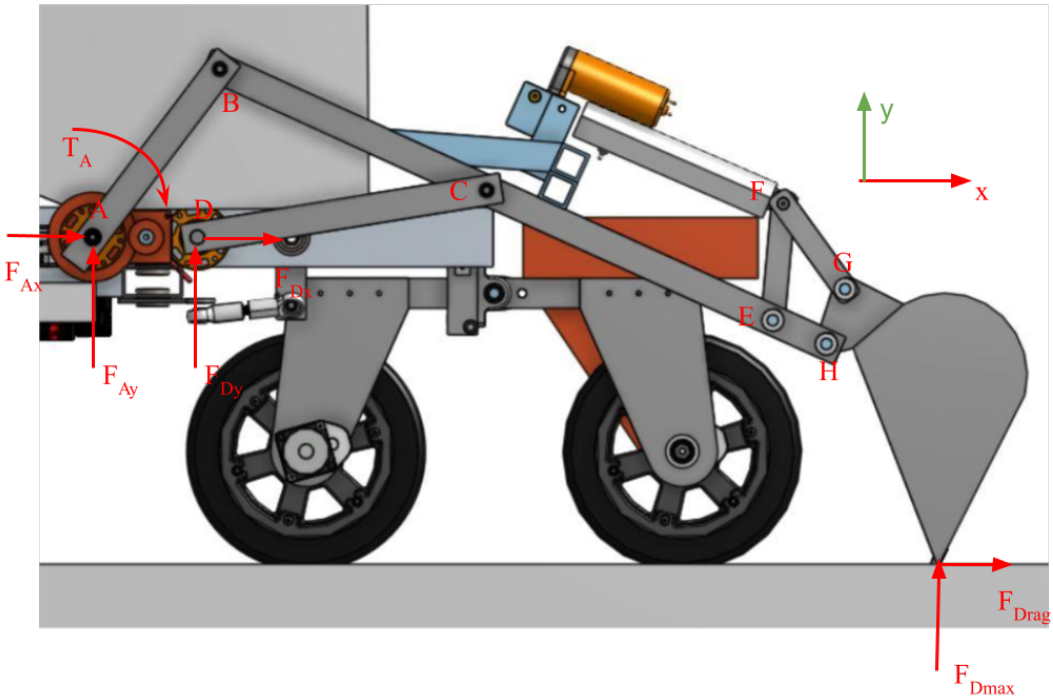

Figure 35: Free body diagram of the excavating mechanism when the maximum digging force occurs

Besides F_{Dmax} and F_{Drag}, forces acting on the digging mechanism include joint forces at joint A and D and the driving torque T_A from the output of the gear train. Since the weight of the digging mechanism will actually help it overcome the resistances, it is fair to neglect the weight of the members and overestimate the required driving torque to ensure a larger margin for safe operation. To determine the required driving torque to overcome

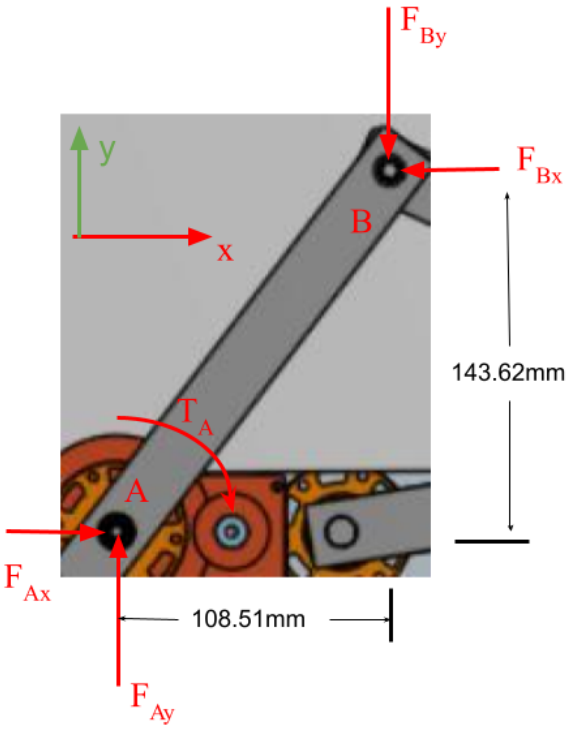

Figure 36: Free body diagram of link pair AB when the maximum digging force occurs

the resistances and the load at each of the joint, each individual member of the digging mechanism also need to analyzed.

in figure 35, F_{Ax}, F_{Ay}, F_{Bx} and F_{By} are horizontal and vertical components of forces at joint A and B. T_A is the driving torque. From the free body diagram of link pair AB, equations of equilibrium can be derived.

$$\sum F_x = F_{Ax} - F_{Bx} = 0 \tag{4}$$

$$\sum F_y = F_{Ay} - F_{By} = 0 \tag{5}$$

$$\sum M_A = F_B x * 143.62mm - F_B y * 108.51mm - T_A = 0 \tag{6}$$

in figure 36, F_{Cx} and F_{Cy} are the horizontal and vertical components of force at joint C. The free body diagram of link BC considers the coupler BCEH, the linear actuator and its back supports as well as the scoop mechanism as an entire system, and equations of equilibrium can be derived.

$$\sum F_x = F_{Bx} + F_{Cx} + F_{Drag} = 0 \tag{7}$$

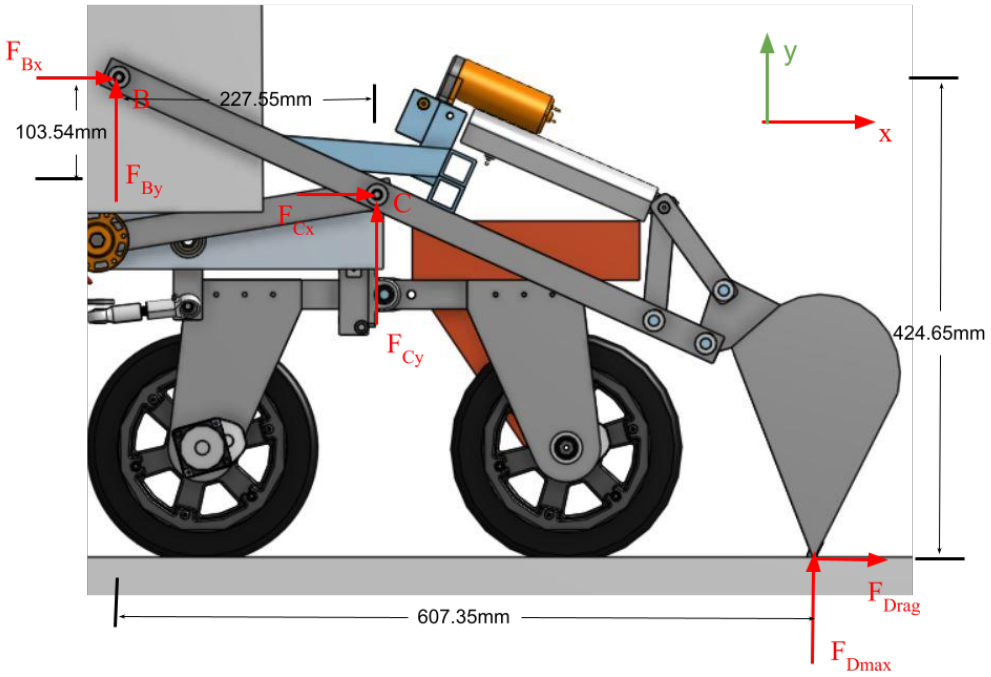

Figure 37: Free body diagram of link BC when the maximum digging force occurs

$$\sum F_y = F_{By} + F_{Cy} + F_{Dmax} = 0 \tag{8}$$

$$\sum M_B = F_{Cx} * 103.54mm + F_{Cy} * 227.55mm + F_{Dmax} * 607.35mm + F_{Drag} * 424.65mm = 0 \tag{9}$$

in figure 37, F_{Dx} and F_{Dy} are the horizontal and vertical components of force at joint D. From the free body diagram of link pair CD, equations of equilibrium can be derived.

$$\sum F_x = F_{Dx} - F_{Cx} = 0 \tag{10}$$

$$\sum F_y = F_{Dy} - F_{Cy} = 0 \tag{11}$$

$$\sum M_D = F_{Cx} * 40.08mm - F_{Cy} * 246.77mm = 0 \tag{12}$$

By solving equations 4 to 12, the results are

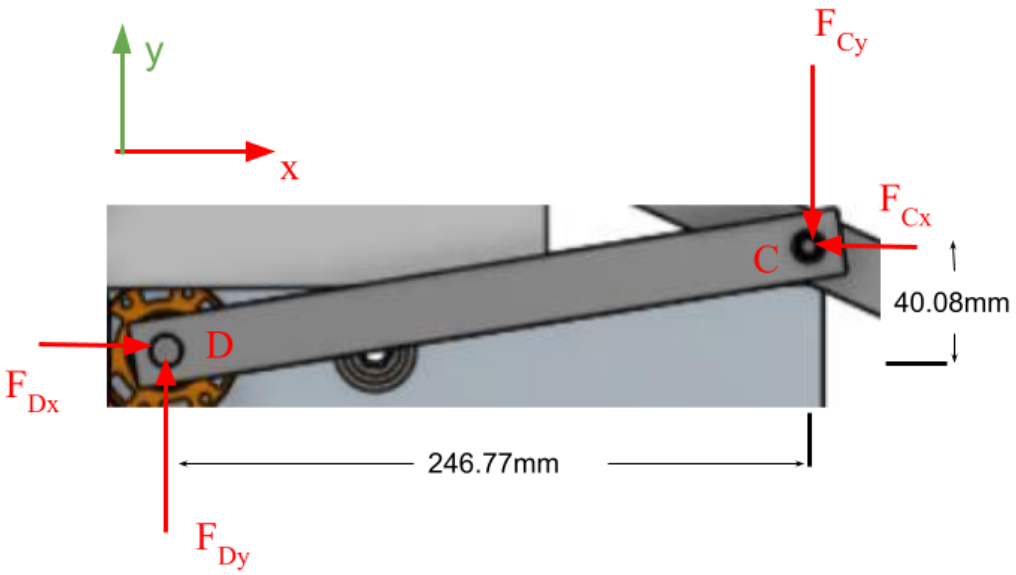

Figure 38: Free body diagram of link pair CD when the maximum digging force occurs

unknown	value
F_{Ax}	717.7N
F_{Ay}	-43.2N
T_A(Digging)	107.8N.m
F_{Bx}	717.7N
F_{By}	-43.2N
F_{Cx}	-727.7N
F_{Cy}	-118.2N
F_{Dx}	-727.7N
F_{Dy}	-118.2N

As shown in the results, the maximum driving torque required to overcome the resistance during digging is $T_A(Digging) = 107.8N.m$. Since the total gear reduction from the BAG motor to the output shaft that drives link pair AB is $2700:1$, the torque load on the BAG motor will be $\frac{107.8N.m}{2700} = 0.04N.m$. And according to the performance curve of the BAG Motor under 12V supply, the motor will be running at around 11900RPM with 6.4A of current and developing around 50W of output power with 64% efficiency.

After examining the torque required to actuate the entire mechanism, it is also important to analyze the scoop mechanism to determine the force required at the linear actuator to actuate the scoop and collect material and loads induced at the joints. Weights of members are also neglected for larger margin of safety.

As shown in figure 39, F_{Ex}, F_{Ey}, F_{Hx}, F_{Hy} are the horizontal and vertical components of joint forces at joint E and H, F_{LD} is the force acting upon joint F from the linear actuator.

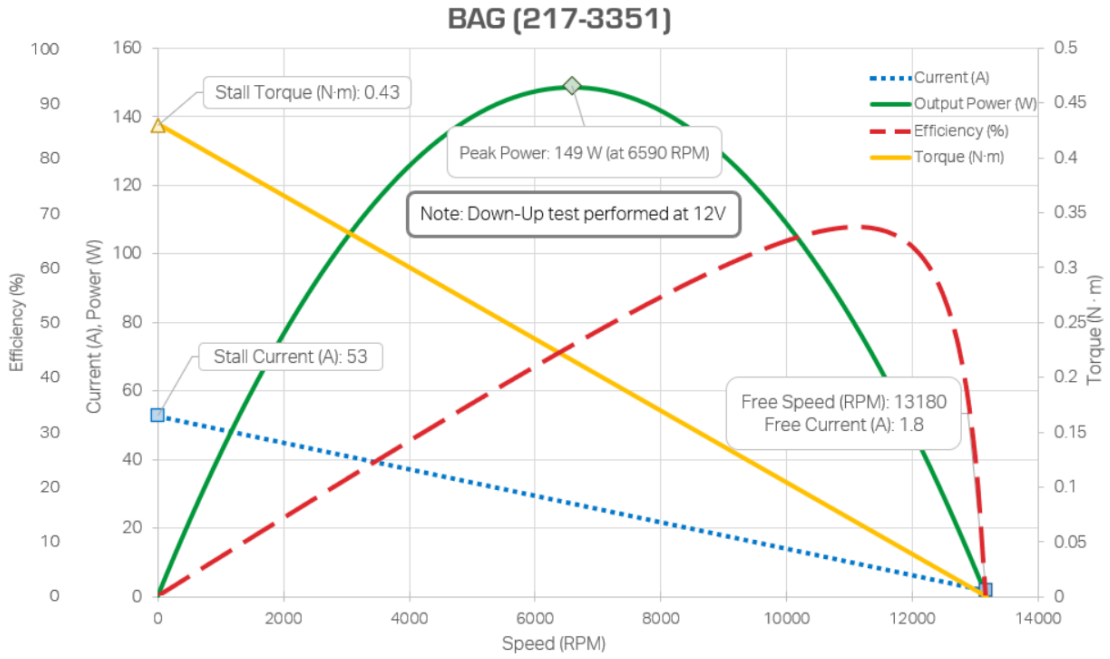

Figure 39: Performance curve of the BAG(217-3351) motor under 12V supply

As shown in figure 40, F_{FEx}, F_{FEy} are the horizontal and vertical components of the force acting upon link pair EF from joint F. From the free body diagram of link pair EF, equations of equilibrium can be derived.

$$\sum F_x = F_{Ex} - F_{FEx} = 0 \tag{13}$$

$$\sum F_y = F_{Ey} - F_{FEy} = 0 \tag{14}$$

$$\sum M_E = F_{FEx} * 99.6mm - FFEy * 8.96mm = 0 \tag{15}$$

As shown in figure 41, F_{FGx}, F_{FGy} are the horizontal and vertical components of the force acting on link pair FG from joint F. F_{Gx}, F_{Gy} are the horizontal and vertical components of the force at joint G. From the free body diagram of link pair FG, equations of equilibrium can be derived.

$$\sum F_x = F_{FGx} - F_{Gx} = 0 \tag{16}$$

$$\sum F_y = F_{FGy} - F_{Gy} = 0 \tag{17}$$

$$\sum M_F = -F_{FGy} * 52.67mm - F_{Gx} * 72.98mm = 0 \tag{18}$$

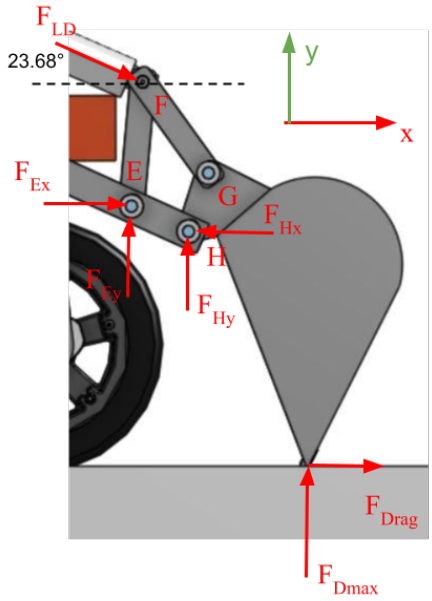

Figure 40: Free body diagram of the scoop mechanism when the maximum digging force occurs

From the free body diagram of the scoop in figure 42, equations of equilibrium can be derived.

$$\sum F_x = F_{Gx} - F_{Hx} = 0 \tag{19}$$

$$\sum F_y = F_{Hy} + F_{Gy} + F_{Dmax} = 0 \tag{20}$$

$$\sum M_H = F_{Gy} * 15.92mm - F_{Gx} * 46.87mm + F_{Dmax} * 95.1mm + F_{Drag} * 189.28mm = 0 \tag{21}$$

As shown in figure 44, F_{LD} is the force acting on joint F from the linear actuator at a 23.68° angle with the horizon. From the free body diagram of joint F, equations of equilibrium can be derived.

$$\sum F_x = F_{FEx} + F_{LD} * \cos(23.68°) - F_{FGx} = 0 \tag{22}$$

$$\sum F_y = F_{FEy} - F_{FGy} - F_{LD} * \sin(23.68°) = 0 \tag{23}$$

by solving equation 13 to 23, the results are

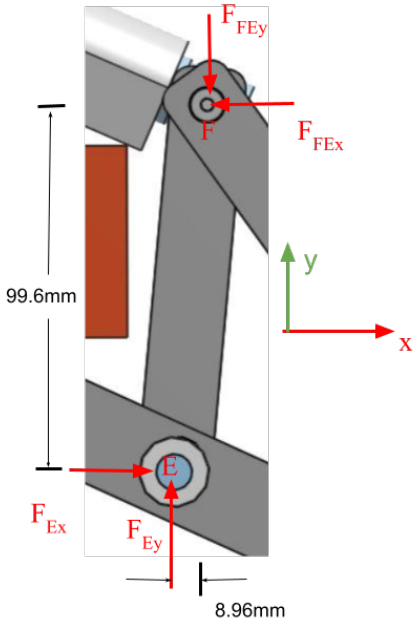

Figure 41: Free body diagram of link pair EF when the maximum digging force occurs

unknown	value
F_{Ex}	-42.1N
F_{Ey}	-468.4N
F_{FEx}	-42.1N
F_{FEy}	-468.4N
F_{Gx}	250.1N
F_{Gy}	-346.5N
F_{FGx}	250.1N
F_{FGy}	-346.5N
F_{Hx}	-250.1N
F_{Hy}	185.1N
F_{LD}	319.1N

According to the results, to overcome resistance and collect material, the load on the linear actuator is $F_{LD} = 319.1N$ which is 24% of its rated static load which is 1334N.

5.1.2 Holding Analysis

Besides digging, another situation that will induce significant load in the excavating mechanism is when it is holding materials collected and delivering it to the collector as

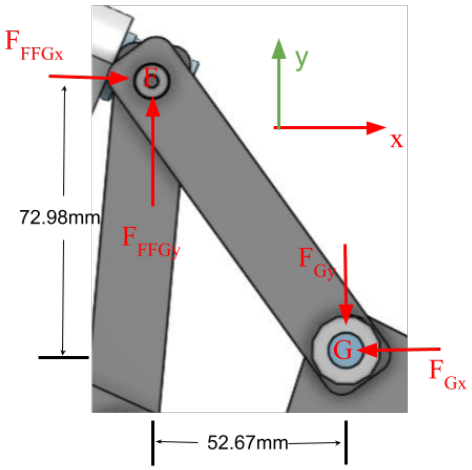

Figure 42: Free body diagram of link pair FG when the maximum digging force occurs

shown in figure 44. Unlike digging, the weight of the members are now part of the resistance to overcome and thus must be considered for analysis. W_{AB}, W_{CD} are the weight of the link pairs AB and CD. W_{BC} is the combined weight of the coupler BCEH, the linear actuator and its supports and the scoop mechanism. W_{MC} is the weight of the material collected.

weight	value
W_{AB}	$2 \times 0.164\text{kg} \times 9.81\text{m/s}^2 = 3.22\text{N}$
$W_C D$	$2 \times 0.207\text{kg} \times 9.81\text{m/s}^2 = 4.06\text{N}$
W_{MC}	$5.4\text{kg} \times 9.81\text{m/s}^2 = 52.97\text{N}$

From the free body diagram of link pair AB in figure 45, equations of equilibrium can be derived.

$$\sum F_x = F_{Bx} - F_{Ax} = 0 \tag{24}$$

$$\sum F_y = F_{By} - F_{Ay} - W_{AB} = 0 \tag{25}$$

$$\sum M_A = F_{By} * 10.6mm - F_{Bx} * 179.69mm - W_{AB} * \frac{10.6mm}{2} + T_A = 0 \tag{26}$$

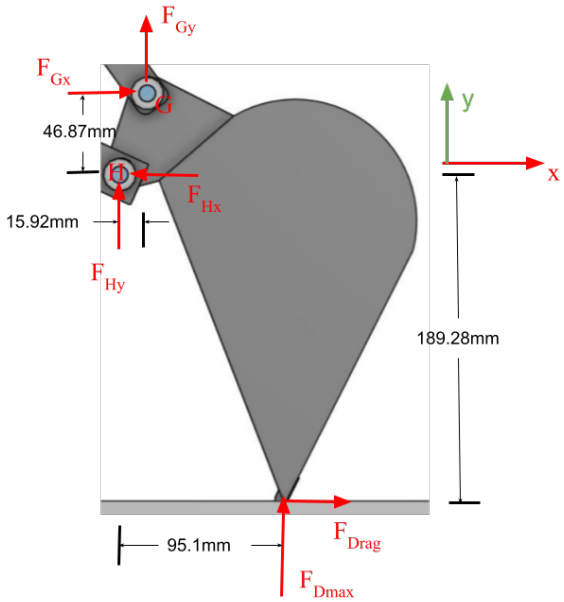

Figure 43: Free body diagram of the scoop when the maximum digging force occurs

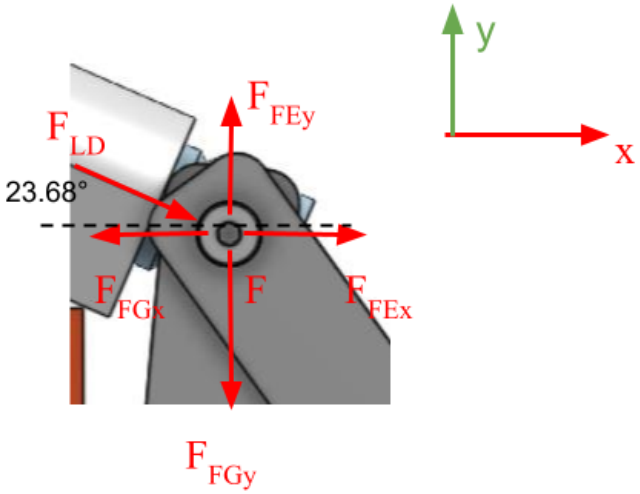

Figure 44: Free body diagram of joint F when the maximum digging force occurs

From the free body diagram of coupler BCEH in figure 46, equations of equilibrium can be derived.

$$\sum F_x = F_{Cx} - F_{Bx} = 0 \tag{27}$$

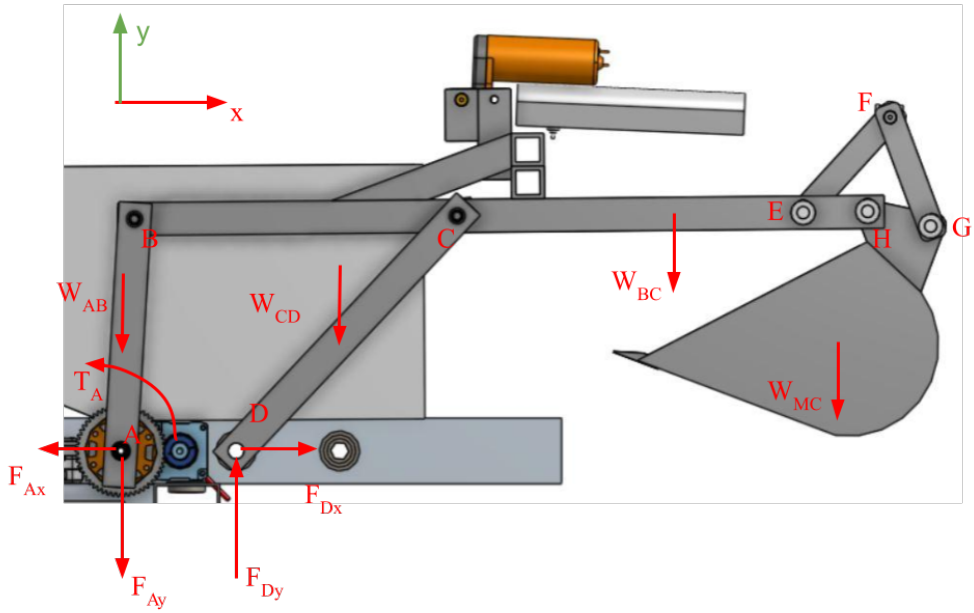

Figure 45: Free body diagram of the excavating mechanism when holding the material collected

$$\sum F_y = F_{Cy} - F_{By} = 0 \tag{28}$$

$$\sum M_B = F_{Cy} * 250mm - W_{BC} * 410mm - W_{MC} * 545mm = 0 \tag{29}$$

From the free body diagram of link pair CD in figure 47, equations of motion can be derived.

$$\sum F_x = F_{Dx} - F_{Cx} = 0 \tag{30}$$

$$\sum F_y = F_{Dy} - F_{Cy} - W_{CD} = 0 \tag{31}$$

$$\sum M_D = F_{Cx} * 182.1mm - F_{Cy} * 171.3mm - W_{CD} * \frac{171.3mm}{2} = 0 \tag{32}$$

By solving equations 24 to 32, the results are,

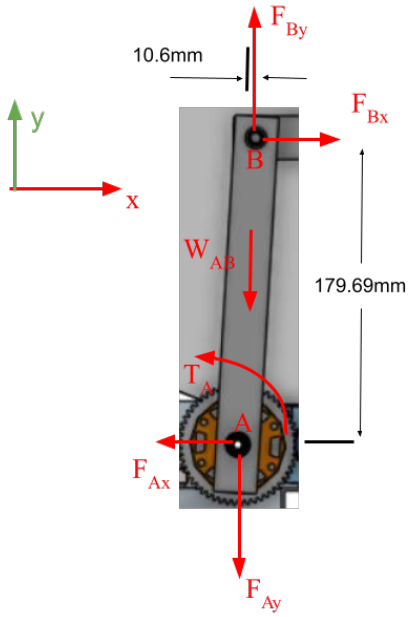

Figure 46: Free body diagram of link pair AB when holding the material collected

unknown	value
F_{Ax}	179.3N
F_{Ay}	87.8N
T_A(Holding)	31.3N.m
F_{Bx}	179.3N
F_{By}	91N
F_{Cx}	179.3N
F_{Cy}	188.5N
F_{Dx}	179.3N
F_{Dy}	192.6N

As shown by the result, the torque required at joint A to hold the weight of the mechanism as well as the material collected is $T_A(Holding) = 31.3N.m$. And the load on the BAG motor through the 2700:1 gear train is $\frac{31.3N.m}{2700} = 0.01N.m$. According to the performance curve of the BAG motor, The motor will be running at around 12800RPM with 3.3A of current, delivering 17W of output power with 40% of efficiency.

As shown in figure 48, the weight of the members in the scoop mechanism is also considered to determine the load on the linear actuator and at the joints. W_{EF}, W_{FG} are the weight of link pairs EF and FG. W_S is the weight of the scoop without material collected. F_{LH} is the driving force from the linear actuator.

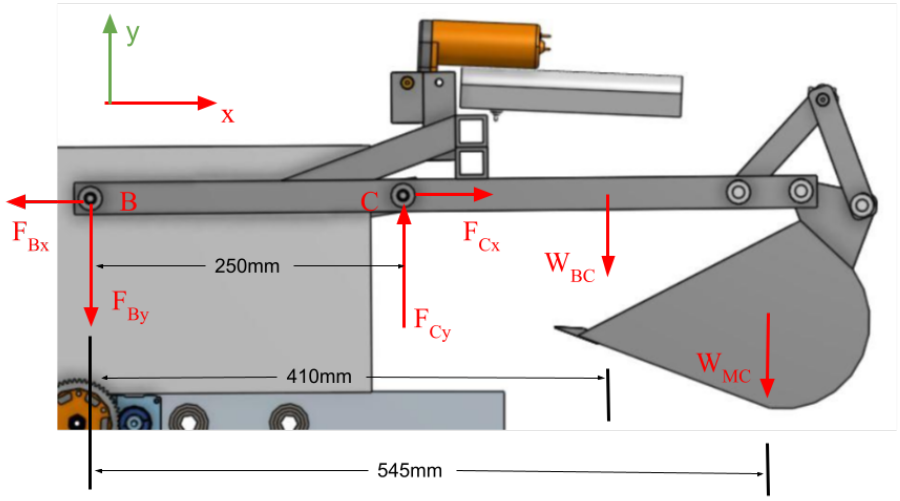

Figure 47: Free body diagram of link BC when holding the material collected

weight	value
W_{EF}	$2 \times 0.0019\text{kg} \times 9.81\text{m/s}^2 = 0.37\text{N}$
W_{FG}	$2 \times 0.017\text{kg} \times 9.81\text{m/s}^2 = 0.33\text{N}$
W_S	$1.636\text{kg} \times 9.81\text{m/s}^2 = 16.05\text{N}$

From the free body diagram of link pair EF in figure 49, equations of equilibrium can be derived.

$$\sum F_x = F_{Ex} - F_{FEx} = 0 \tag{33}$$

$$\sum F_y = F_{Ey} - F_{FEy} - W_{EF} = 0 \tag{34}$$

$$\sum M_E = F_{FEx} * 31.4mm - F_{FEy} * 67.4mm - W_{EF} * \frac{67.4mm}{2} = 0 \tag{35}$$

From the free body diagram of link pair FG in figure 50, equations of equilibrium can be derived.

$$\sum F_x = F_{FGx} - F_{Gx} = 0 \tag{36}$$

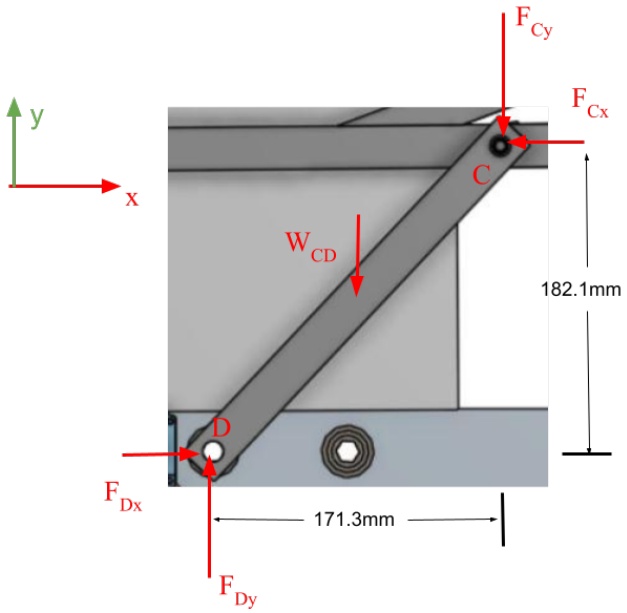

Figure 48: Free body diagram of link pair CD when holding the material collected

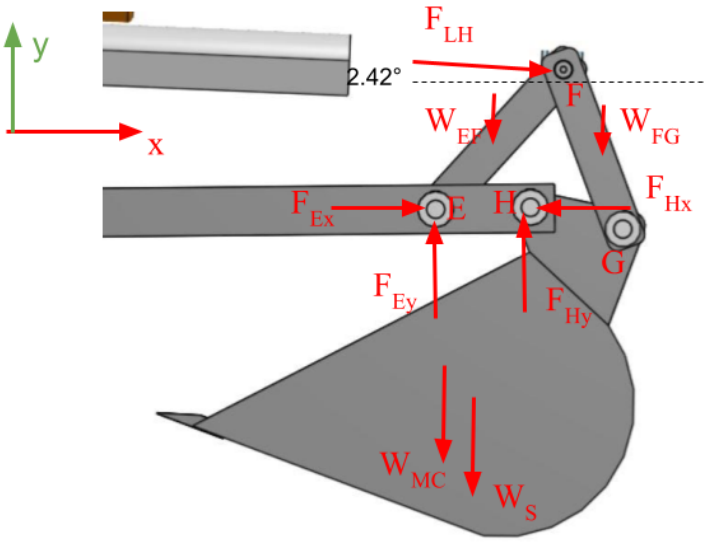

Figure 49: Free body diagram of the scoop mechanism when holding the material collected

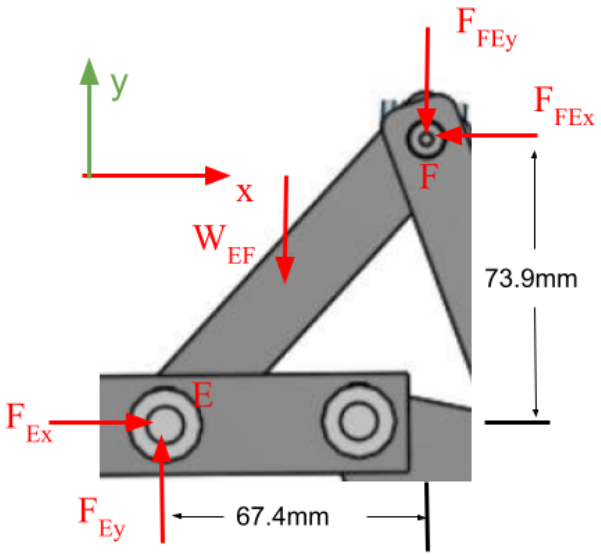

Figure 50: Free body diagram of link pair EF when holding the material collected

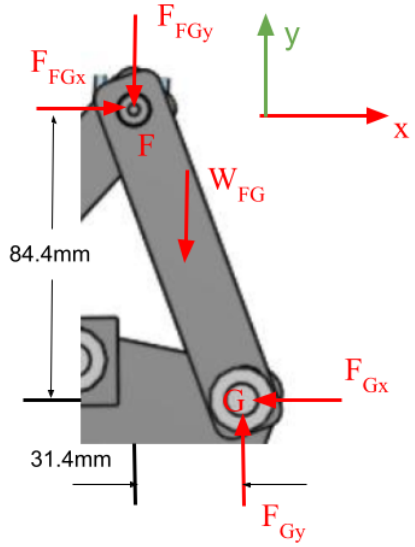

Figure 51: Free body diagram of link pair FG when holding the material collected

$$\sum F_y = F_{Gy} - F_{FGy} - W_{FG} = 0 \tag{37}$$

$$\sum M_F = F_{Gy} * 31.4mm - F_{Gx} * 84.4mm - W_{FG} * \frac{31.4mm}{2} = 0 \tag{38}$$

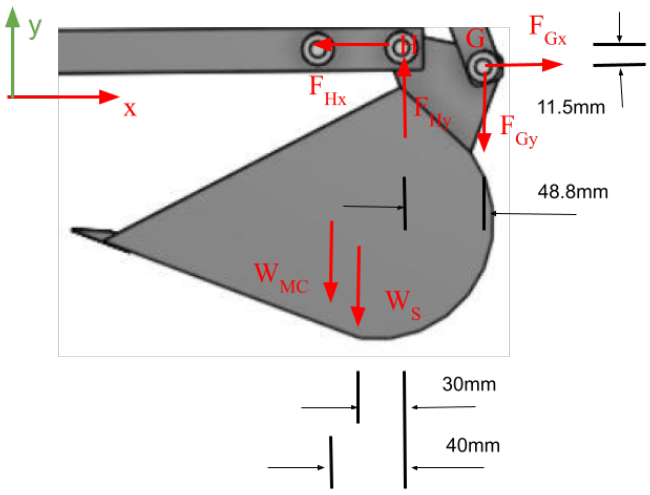

Figure 52: Free body diagram of the scoop when holding the material collected

From the free body diagram of the scoop in figure 51, equations of equilibrium can be derived.

$$\sum F_x = F_{Gx} - F_{Hx} = 0 \tag{39}$$

$$\sum F_y = F_{Hy} - F_{Gy} - W_{MC} - W_S = 0 \tag{40}$$

$$\sum M_H = F_{Gx} * 11.5mm - F_{Gy} * 48.8mm + W_S * 30mm + W_{MC} * 40mm = 0 \tag{41}$$

From the free body diagram of joint F, equations of equilibrium can be derived.

$$\sum F_x = F_{FEx} - FFGx + F_{LH} * \cos(2.42°) = 0 \tag{42}$$

$$\sum F_y = F_{FEy} + F_{FGy} - F_{LH} * \sin(2.42°) = 0 \tag{43}$$

And by solving equation 33 to 43, the results are,

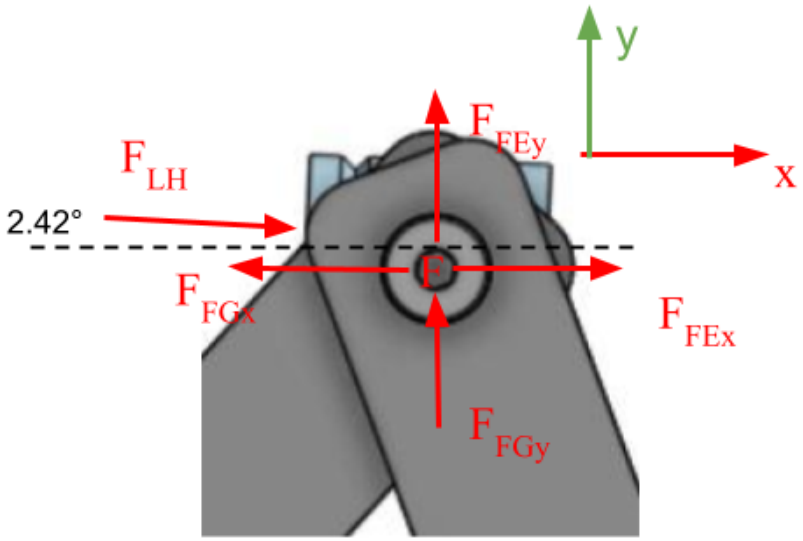

Figure 53: Free body diagram of joint F when holding the material collected

unknown	value
F_{Ex}	-50.5N
F_{Ey}	-55.2N
F_{FEx}	-50.5N
F_{FEy}	-55.5N
F_{Gx}	21.9N
F_{Gy}	58.9N
F_{FGx}	21.9N
F_{FGy}	58.6N
F_{Hx}	21.9N
F_{Hy}	127.9N
F_{LH}	72.4N

according to the results, the required load on the linear actuator to hold the material collected is $F_{LH} = 72.4N$, which is 5.4% of its rated static load which is 1334N.

5.1.3 Rocker-Bogie Analysis

A situation that will impose significant load in the Rocker-Bogie driving system of the robot is when it is climbing over obstacles and lifting its own weight. The analysis is performed assuming that the robot is climbing over a 20cm high, 50cm long plateau. As shown in figure 5.1.3, W_R is the weight of the robot. F_{nR}, F_{nM}, F_{nF} are the normal forces between the front, middle and back wheels and the surfaces of contact. F_{fR}, F_{fM}, F_{fF} are the frictional forces generated at the contact surfaces to propel the robot. And their relations can be expressed

by equation 44, where F_f is the frictional force and F_n is the normal force. μ is the coefficient of friction between the surfaces of contact. And the μ between the wheels and the ground is $\mu_s = 0.6$ as that between rubber thread-less tire and sand road Whereas the μ between the wheels and the obstacle is $\mu_o = 0.65$ as that between rubber thread-less tire and dry asphalt. For a larger margin of safety, it is reasonable to assume that the robot is carrying material collected to its full capacity in the collector bucket during climbing. And the weight of the robot when it is fully loaded with sand-gravel mixture is $W_R = 54\text{kg} \times 9.81\text{m/s}^2 = 529.74\text{N}$. And the front is in the same direction as the positive x.

$$F_f = \mu \times F_n \tag{44}$$

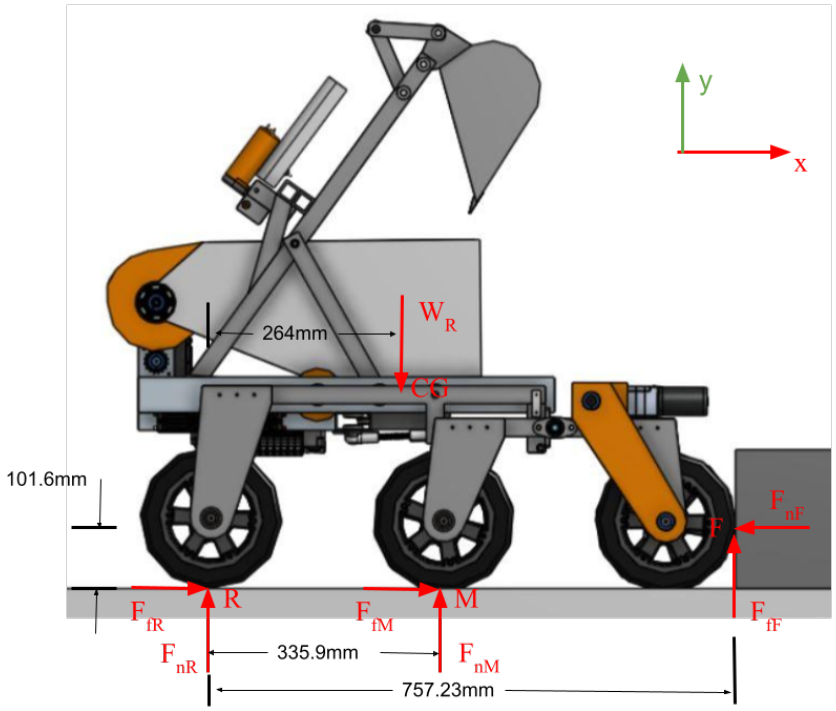

Figure 54: Free body diagram of the robot when the front wheels start climbing onto the obstacle

When the robot starts climbing with it front wheels, the excavating mechanism is fully retracted to locate the CG of the robot backward to minimize the load required at the front wheels for lifting up the robot. And from figure 5.1.3, equations of equilibrium can be derived.

$$\sum F_x = F_{nR} * \mu_s + F_{nM} * \mu_s - F_{nF} = 0 \tag{45}$$

$$\sum F_y = F_{nR} + F_{nM} + F_{nF} * \mu_o - W_R = 0 \tag{46}$$

$$\sum M_R = F_{nM} * 335.9mm + F_{nF} * (\mu_o * 757.23mm + 101.6mm) - W_R * 264mm = 0 \quad (47)$$

And by solving equation 45 to 47, the results are,

unknown	value
F_{nR}	369N
F_{nM}	12.1N
F_{nF}	228.7N
F_{fR}	221.4N
F_{fM}	7.3N
F_{fF}	148.6N

The results indicates that the highest normal and torque loads are at the rear wheels when the front wheels start climbing onto the obstacle. To determine the torque load at each of the wheels, equation 48 is used. T_{wheel} is the torque load on a single driving wheel in N.m, $F_{fWheelPair}$ is the frictional force generated at the front, middle or the back wheels pair in N, and D_wheel is the diameter of the driving wheels in millimeter which is 203.2mm for the rocker-bogie.

$$T_{wheel} = \frac{F_{fWheelPair}}{2} \times \frac{D_{wheel}}{2} \times \frac{1m}{1000mm} \quad (48)$$

And the largest torque load on the wheels in this case would be $T_{wheel} = \frac{F_{fR}}{2} \times \frac{203.2mm}{2} \times \frac{1m}{1000mm} = 11.25N.m$. The load on the driving BAG motor through the 100:1 planetary gear reduction will be $T_{motor} = \frac{11.25N.m}{100} = 0.11N.m$. According to the performance of the BAG motor, the motor will be running at around 9800RPM with 15A of current, delivering 114W of output power with a 64% efficiency.

.

Figure 5.1.3 shows the robot with a 8.5° pitch angle when its middle wheels start climbing onto the obstacle. From the free body diagram of the robot, equations of equilibrium can be derived.

$$\sum F_x = F_{nR} * \mu_s - F_{nM} + F_{nF} * \mu_o = 0 \quad (49)$$

$$\sum F_y = F_{nR} + F_{nM} * \mu_o + F_{nF} - W_R = 0 \quad (50)$$

$$\sum M_R = F_{nM}*(101.6mm+\mu_o*527.5mm)+F_{nF}*(681.4mm-\mu_o*200mm)-W_R*238.4mm = 0 \quad (51)$$

And by solving equation 49 to 51, the results are,

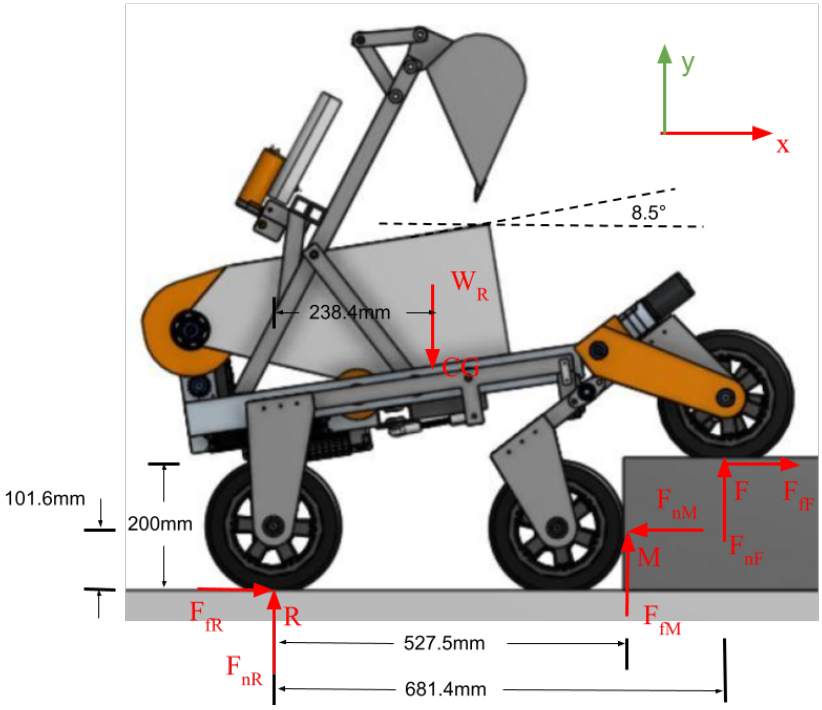

Figure 55: Free body diagram of the robot when the middle wheels start climbing onto the obstacle

unknown	value
F_{nR}	336.4N
F_{nM}	228.7N
F_{nF}	44.7N
F_{fR}	201.8N
F_{fM}	137.2N
F_{fF}	29.1N

The results indicate that when the middle wheels of the robot start climbing onto the obstacle, the rear wheels still experience the most normal and torque load. And the torque load on each of the rear wheels is $T_{wheel} = \frac{F_{fR}}{2} \times \frac{203.2mm}{2} \times \frac{1m}{1000mm} = 10.25N.m$. And the torque load on the BAG motor through the 100:1 planetary gear reduction will be $T_{motor} = \frac{10.25N.m}{100} = 0.1N.m$. According to its performance curve, it will be running at about 10100RPM with 13.5A of current, delivering 105W of output power with a 65% efficiency.

.

Figure 5.1.3 shows the robot with a 25.5° pitch angle when its rear wheels start climbing onto the obstacle. The excavating mechanism is extended forward to move the CG of the robot towards the front for both stabilizing the robot during climbing and reducing the loads required at the rear wheels to lift the robot up. From the free body diagram of the robot at

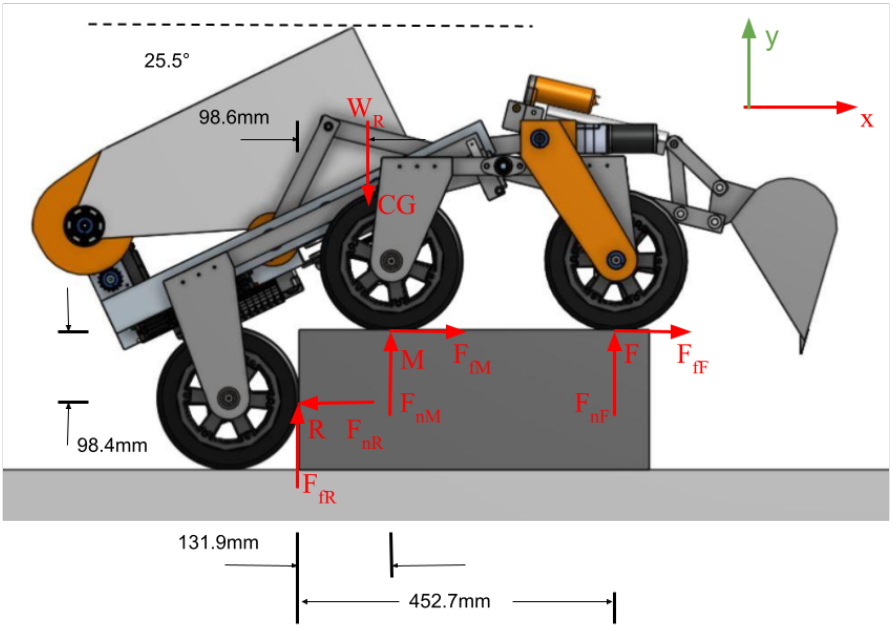

Figure 56: Free body diagram of the robot when the rear wheels start climbing onto the obstacle

this instance, equations of equilibrium can be derived

$$\sum F_x = -F_{nR} + F_{nM} * \mu_o + F_{nF} * \mu_o = 0 \tag{52}$$

$$\sum F_y = F_{nR} * \mu_o + F_{nM} + F_{nF} - W_R = 0 \tag{53}$$

$$\sum M_R = F_{nM} * (131.9mm - \mu_o * 98.4mm) + F_{nF} * (452.7mm - \mu_o * 98.4mm) - W_R * 98.6mm = 0 \tag{54}$$

By solving equation 52 to 55, the results are,

unknown	value
F_{nR}	242.1N
F_{nM}	288.4N
F_{nF}	84N
F_{fR}	145.2N
F_{fM}	173.1N
F_{fF}	54.6N

The results indicates that when the rear wheels of the robot start climbing onto the obstacle, the middle wheels experience the most normal and torque load. And the torque load on each of the middle wheels is $T_{wheel} = \frac{F_{fM}}{2} \times \frac{203.2mm}{2} \times \frac{1m}{1000mm} = 8.79N.m$. And the torque load on the BAG motor through the 100:1 planetary gear reduction will be $T_{motor} = \frac{8.79N.m}{100} = 0.088N.m$. According to its performance curve, it will be running at about 10500RPM with 12A of current, delivering 95W of output power with a 66% efficiency.

5.1.4 Dumping Analysis

When the collector bucket is fully loaded and driven to its dumping position, due to the heavy weight of the material collected (sand-gravel mixture), the CG of the robot will be shifted upwards and towards the rear side of the robot. In order to keep the robot from tipping backwards during dumping and maintain even loads at the wheels, the excavating mechanism is deployed forward as shown in figure 5.1.4. The vertical projection of the resulted CG of still remains in the polygon formed by the contact points between the wheels and the ground, thus keeping the robot stable.

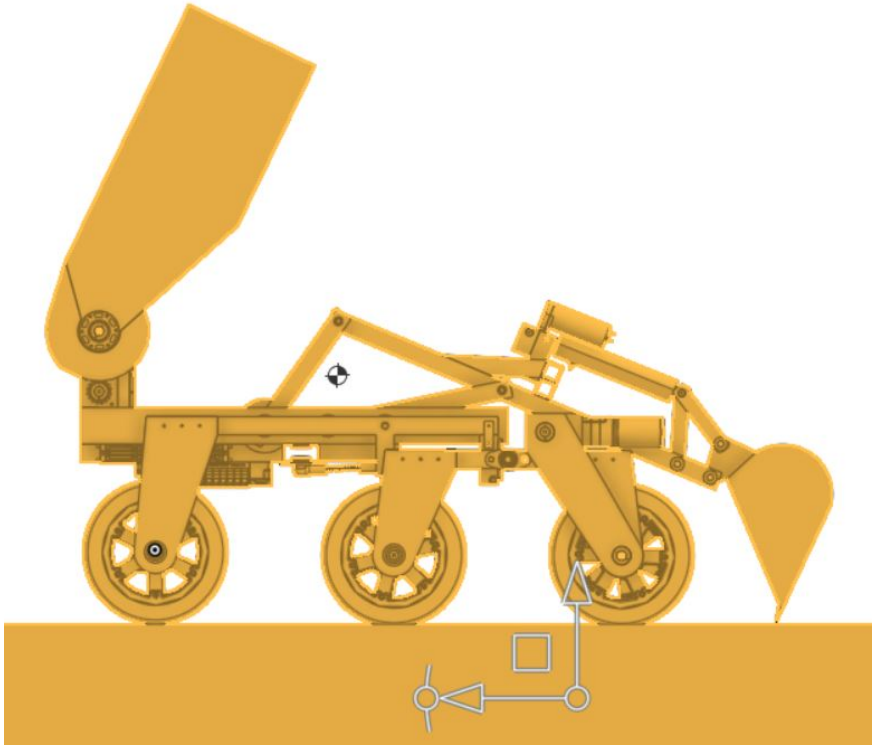

Figure 57: CG of the robot during dumping configuration

.

The maximum torque required to lift the collector occurs right after the collector loses contact with its support on the chassis as shown in figure 5.1.4 where the moment arms of

the resisting weights are the longest relative to joint P. The weight of the polycarbonate collector W_C is 1.8kg $\times$ 9.81m/s^2 = 17.66N. The weight of the sand-gravel mixture stored in the collector when fully loaded W_{MS} is 14kg $\times$ 9.81m/s^2 = 137.34N. T_P is the driving torque at joint P. From the free body diagram of the bucket, equations of equilibrium can be derived.

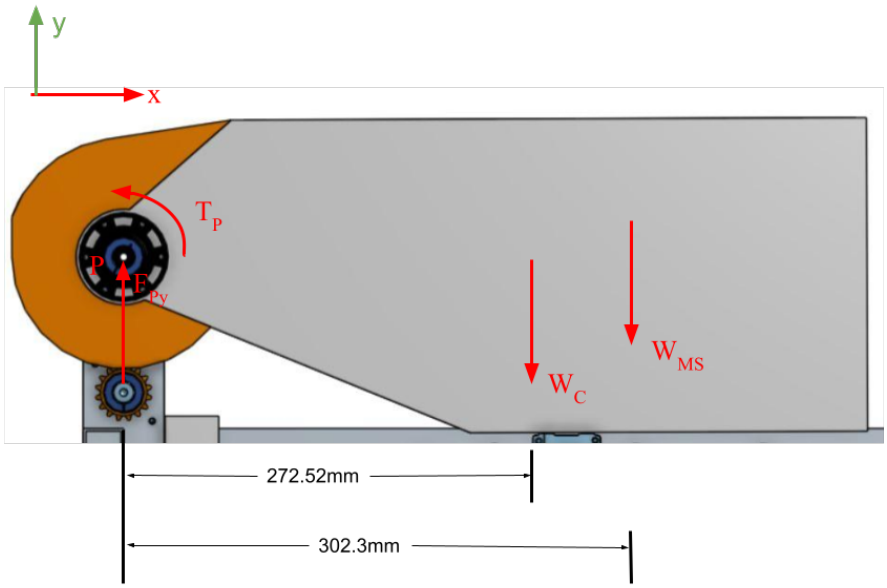

Figure 58: Free body diagram of the collector bucket at the start of dumping

.

$$\sum F_y = F_{Py} - W_C - W_{MS} = 0 \tag{55}$$

$$\sum M_P = T_P - W_C * 272.52mm - W_{MS} * 302.3mm = 0 \tag{56}$$

Solving equations 55 and 56, the results are,

unknown	value
F_{Py}	155.1N
T_P	46.3N.m

And the load on the BAG motor through the 400:1 gear reduction is $T_{motor} = \frac{46.3N.m}{400} = 0.116N.m$. According to motor performance, it will be running at around 9700RPM with 15.5A of current, delivering 115W of output power with a 63% efficiency.

5.2 Gear Calculations

To power the excavator four-bar, the dumping bin, and the wheels, the team used Vex bag motors with gearboxes corresponding to the subsystem's needs. For the four-bar and dumping container, there were two requirements that needed to be met. These were the max load they would carry and the time it would take for full range of motion. The following equations were used to calculate the gear speed ratio i, output speed w_{out}, and output torque T_{out} [10]. To calculate the gear ratio i for spur gears, the number of teeth N were divided as shown in equation 57. The ratios for the planetary stages were given.

$$i = N_{in}/N_{out} \tag{57}$$

The output speed $w - out$ was calculated by multiplying the gear ratios with the input speed in equation 58. The input speed w_{in} was taken from the FRC Bag motor operating at maximum efficiency in the motor curve [11].

$$w_{out} = w_{in} * 1/i_1 * 1/i_2 * 1/i_3 \tag{58}$$

The output torque T_{out} was calculated in equation 59 assuming an efficiency of .97 for each planetary stage and .99 for the spur gears. The input torque T_{in} was taken from the FRC bag motor operating at maximum efficiency [11].

$$T_{out} = T_{in} * \eta_1 * i_1 * \eta_2 * i_2 * \eta_3 * i_3 \tag{59}$$

For the dumping action, the max load is calculated to be 64.1 Nm. A 100:1 two stage planetary gear box and a chain and sprocket with a 4:1 ratio would give the desired motion in 5 seconds. Lastly, the wheels have a very different role and requirements. The optimal way to drive on BP-1 is slowly and at a steady speed. Furthermore, there will also be the possibility of climbing over small obstacles. This system requires low speed and high torque. This is achieved by 6 independently powered wheels each with a two stage 89:1 planetary gearbox. This results in the wheels having a max rotation of 172.8 rpm, or a top speed of about 3.8 meters per second.

For the four-bar, the max load was calculated to be 18.36 Nm. With this in consideration, the team chose to use a 100:1 gearbox with 3:1 spur gears, resulting in a 300:1 gear ratio. This configuration allowed the full range of motion for the excavator to be completed in under 2 seconds with a carrying capacity of 109.37 Nm. However, once the excavator was actually built, it struggled to lift a scoop of material. An additional 9:1 gear stage was added to the planetary gearbox, bringing the gear ratio to 2700:1. The revised calculations can be found in the sections above.

5.3 Power Calculations

One of the key components of the robot is the battery and power system. The battery is one of the heaviest parts of the robot. Selecting the right size of battery is very important

- too large of a battery and there is excessive mass. If the battery is too small, the robot may stop moving during the competition. To estimate the amount of power required by one round of competition, the various demands on the battery were estimated and summed up in the table below.

Component	Est. Power Draw	Time	Total
6 Drive Motors	10 A	5 min	5 AH
Excavator Four-Bar	15 A	10 min	2.9 AH
Excavator Linear Actuator	5 A	10 min	.83 AH
Dump Bucket	15 A	2 min	.5 AH
Electronics	4 A	15 min	1 AH
Sum:			9 AH

The demands on the motors were estimated from the gear calculations and expected loads on the motors. For the electronics and linear actuator, the maximum steady current was used. The time was based on the amount of time the robot would use each mechanism within the competition. Based on this table, the team selected a 20 Amp-Hour battery as the most suitable choice for the robot. That gives the team the ability to keep the battery charge in the optimum range for maintaining a Lithium battery as well as the potential to do some light testing before a competition round. Ideally, two batteries would be purchased and could be swapped out at need.

5.4 Excavator Performance

After testing the efficiency of the excavator, it was found that a single scoop averaged 3.5 lbs (1.6 kg) of sand. It took 23 scoops to reach the gravel layer at 30 cm deep in teleoperation mode. Once at the gravel layer it took four scoops to dig up 3 lbs (1.36 kg) of gravel. Despite the gravel used for testing being about twice the size of the competition's icy regolith, there was no difficulty in collecting it. The testing was also completed using only a few layers of gravel. In the competition, the regolith layer consists of 15 centimeters of gravel that is 30 centimeters below the surface. The team believes that the excavator would perform better under the actual competition specifications due to the smaller gravel site and higher quantity of gravel. Once the testing was completed with teleoperation the autonomous system was implemented and tested (setup shown in figure 59. The excavator was able to complete a full cycle of digging in seven minutes. It took 9 scoops to complete the cycle.

It was determined by the team that to maximize the digging potential of the excavator, the robot should be moved a few millimeters each time the scoop dumps its load into the bucket. This allows for a favorable angle of attack as the scoop needs to go deeper and deeper into the ground with each pass. This can be achieved by having the robot drive backwards between scooping cycles. Overall the excavator performed well and with some minor adjustments the team believes that this design can be utilized for future Lunabotics competitions.

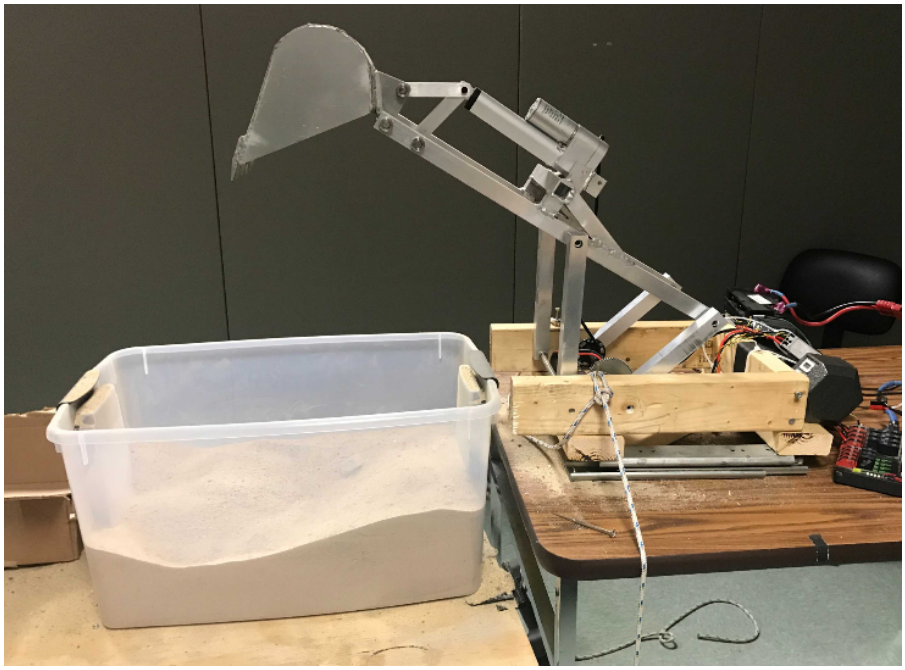

Figure 59: Excavator Test Setup

To the team's surprise, the horizontal force experienced at the base of the excavator due to the digging process was very consistent. On all but two of the scoop cycles the horizontal force was 10 Newtons towards the front of the robot. This force should be carefully considered when determining the material of the wheels. If the friction force between the wheels and the ground does not exceed 10 Newtons then the robot will be dragged forward during the digging process. This force did not move the the wooden base substantially. Due to limited time and resources, tests on the friction force between the wheels and the ground were not performed. However, the team believes that this transverse force would not be enough to pull the robot forward. As the excavator collects more and more material, the weight of the overall robot will increase. This will result in less movement as the digging continues. If this force were to still pull the robot forward, a simple solution to this problem would be to drive the wheels in reverse.

During the testing phase, there was only one major issue that the team faced. The first tests were done with a gear ratio of 300:1. This ratio did not provide sufficient torque to operate the excavator at its maximum depth and the gears began to skip. A 9:1 gear stage was added to the motor to improve the torque. This brought the total gear ratio to 2700:1. This mostly solved the problem of the gears skipping. Occasionally the gears still skipped but this was not do to lack of torque. The team believes that the are two reasons for this problem. The first is a result of the wooden base not being secure enough for the motor attachment. The screw holes in the 2x4 used to fasten the motor were not strong or precise enough. This caused the motor and gear box to slightly lift off of the wooden base resulting in the skipping gears. Another reason for the skipping was the aluminum hex shaft.

During the digging process the shaft would bend ever so slightly which also contributed to the skipping. The team believes that there are numerous solutions to this problem. The first is to make the base out of aluminum as in the full CAD. The second and best solution to the problem is to use a chain and sprocket system rather than gears. A chain and sprocket system would not have a problem with the high torque required to operate the excavator. The third solution is to use a steel hex shaft rather than an aluminum one. The additional strength that steel provides would be adequate enough to prevent the shaft from bending.

It should also be noted that during the digging process there was a moment created between the scoop and the back of the robot. This moment caused the center of the robot to experience an upward force that could cause the front wheels to lift off the ground. Due to the lack of time and resources the team was not able to accurately determine the force caused by this moment. The wooden base constructed by the team would not have provided an adequate simulation of this moment that the robot would have felt if the chassis and rocker bogie had been constructed. To counteract the moment experienced by the excavator and wooden base, the team hung two ten pound weights, totaling twenty pounds, from each side of the front of the wooden base. This weight kept the base from rising in the front. The team believes that this would not be an issue if the full robot design was constructed but it should be considered if future teams choose to adapt and/or adopt our excavator design to their own.

5.5 Software Performance

The performance of the software built for this MQP was evaluated primarily on the effectiveness of the state machine as well as the control system used in operating the autonomous portion of the digging routine. Teleoperation was also tested to ensure manual control over the excavation system was sufficient. The state machine was evaluated based on its ability to correctly transition from state to state in the correct order we designed for our digging routine. The control system was evaluated based on its ability to accurately reach the designated target position as well as test its ability to react to high current draw events such as reaching digging material early.

5.5.1 State Machine Performance

During the autonomous portion of our testing phase, the state machine proved to work very well and performed close to our expectations. The state machine was able to successfully handle each autonomous cycle which allowed for consecutive cycles to be completed without any major problems. A complete excavation routine was tested with the autonomous system to determine whether the state machine could handle slight variations in sensor feedback as well as overall digging progression. The results of this test indicated that even as the robot dug progressively deeper, which resulted in slightly longer cycles times, the state machine was able to operate as intended regardless of how much time each state required. This meant that the state machine operated solely on the incoming data and then checking to see if transitions were available rather than relying on an arbitrary time estimate. Pictured in

figure 60, each full cycle of the state machine (i.e. the process of collecting a single scoop of material and dumping it), required roughly 40-45 seconds, and adding 2-3 seconds with later depths. The time required to reach a depth of roughly 30 cm required 7-9 full cycles (5-7 minutes) of autonomous excavation. Given the round duration in the competition being 15 minutes, this would allow our robot to collect at least 2 full loads of gravel each round, satisfying our initial material collection goal.

State	Time (s)
DIG_EXTEND_FOURBAR	9
DIG_RETRACT_FOURBAR	8
DIG_EXTEND_SCOOP	13
DUMP_SCOOP	12

Figure 60: Average Time Spent in Each State

5.5.2 Control System Performance

For the control system performance, we observed that its ability to operate was satisfactory given that maintained an error rate of $\leq 20\%$ in states without early transitions (i.e. extending the four-bar) and thus able to keep an accurate feedback loop over multiple autonomous cycles without throwing off the system. Because of this, the control system was able to successfully supply the state machine with accurate feedback and prevent any error states from being reached during autonomous operation. The control system also prevented any motor burnout via current monitoring feedback which influenced when to transition from the four-bar extension phase to the scoop extension phase, thus satisfying another key goal of performing autonomously.

As evident from figure 61, the linear actuator feedback control worked extremely well with $\leq 2\%$ error rate, primarily because of its tighter control, slower movement, and smaller range of motion. Additionally, the feedback mechanism for the linear actuator is independent of the four-bar feedback system and thus no accumulated error from either set of states would influence the other. The four-bar experienced more volatility in its feedback system as the system encountered occasional jumps in the magnetic encoder output as mentioned above

State	Error %
DIG_EXTEND_FOURBAR	12.4%
DIG_RETRACT_FOURBAR	18.3%
DIG_EXTEND_SCOOP	0.05%
DUMP_SCOOP	2%

Figure 61: State Error Rate

which affected the sensor reading by up to roughly 20%. This was believed to have been caused by the quality of the wooden base and gearbox parts. This then caused the four-bar to drop faster than intended and throw off the control system slightly. However, even with this higher error rate, it primarily only affected the ending dump position as noted by the `DIG_RETRACT_FOURBAR` state having the highest error rate. The result of this error manifested as the dumping position occurring slightly farther away from the correct target position as the accumulated error from the previous four-bar actuation state misled the state machine into thinking it had moved the four-bar closer to its reset position. This accumulated error had little affect on the performance of the digging process as the extending phase almost always would have to transition early since it would reach material before its target position and thus never overshoot. As mentioned earlier, this error in the magnetic encoder output is most likely due to the mechanical issue of gear skipping, and can be resolved if the gear skipping is fixed in a future iteration of the project.

5.5.3 Teleoperation Performance

Lastly, the performance of the teleoperation mode was also measured and evaluated to ensure that manual operation of the digging system was enough to debug, operate, and rescue the system. As mentioned in the design section, a controller was used with multiple actuation settings to independently control both the linear actuator and the bag motor. The smoothing factor applied to the motor controls worked well to ease both systems through their range of motions to demonstrate and explore their basic functionality. Additionally, both the output methods: percent and absolute, worked in setting the correct position and/or motor output value and were able to be used easily in the manual testing portion of the project.

6 Project Organization

The organization and management of the team and available resources is crucial to the overall success any endeavor. Project management was more important than ever this year, with COVID-19 having a large impact on what we could and could not do. To cope with these challenges and organize our project, the team used a number of project management tools.

6.1 Tools Utilized

The primary means of communication used by this MQP was the messaging platform Slack. We created a workspace and made channels to post interesting design ideas, progress photos of our robot, and important links. We also coordinated work and messaged each other through the platform, allowing communication for this project to be focused in one application. Our documents and presentations were kept in a shared Google Drive with the exception of this paper, which was written with Overleaf (an online LaTeX editor). Our robot was designed in Onshape, a web-based CAD software. It has many of the features and a similar layout to Solidworks without requiring a Windows operating system or a particularly capable graphics card.

Figure 62: What is scrum?

Our team held weekly meetings with our advisors over Zoom, creating a presentation to show our progress and receive feedback. We also used Zoom for all of our team meetings. In

terms where our schedules allowed (B and D terms) we held daily scrum meetings. Scrum is a project organization tool first popularized in software development where each team member reports on what they have done since the last meeting, what they are going to do before the next meeting, and if they have encountered significant obstacles. It helps to keep each member accountable and provides an easy way to seek help with problems. In the terms when we did not have scrum, we still met four times a week to discuss the status of the project. Overall, the terms with a daily scrum were more productive than those without. In combination with frequent meetings, we had a project management board. Using a web-based tool called Jira, we maintained a Kanban board with tasks and subtasks that needed to be completed. Also borrowed from software engineering, these tasks were sorted into 'backlog', 'selected for development', 'in progress', or 'completed' columns. They were also assigned to individuals, so people could check what tasks they were supposed to be working on or claim new tasks when they completed their work.

6.2 Impact of COVID-19

Like everything else in the past year, Covid has had a significant impact on our project. This section is intended not for excuses or complaints, but to document some of the challenges the team dealt with. There were a number of first order and second order impacts from this global pandemic.

One first order impact was WPI's additional restrictions on lab use and in-person meetings. From the beginning of this project, it was unclear if the school would stay open and if any in-person work could be completed. This had a number of ramifications for our project. First, all of our meetings were done virtually which affected the team dynamic. We chose to do as much online as possible to minimize the risks to ourselves and others using the lab spaces. As when any new technology is adopted, there was a bit of a learning curve in how to best adapt to doing a project primarily over a virtual platform. It is no coincidence that most of our project tools above are web-based; part of our reason for choosing them was that we could use these tools from our personal machines regardless of location. Second, the team made a decision to focus on the CAD before beginning to build any part of our robot. When lab spaces began opening up, parts of the design were still being tweaked and the team could not immediately transition. The team also chose to order everything online as opposed to dropping by a hardware store or searching for spare parts in a lab. This increased both the cost and time required to build the robot.

Another set of impacts from Covid were our challenges in getting and maintaining card access to labs throughout the year. This was partly due to WPI's many campus-wide restrictions. It took the team multiple weeks to be allowed into the labs. Commuter students were initially not given card access, and it took an addition week to complete onboard tests after the issue was resolved. These choices allowed us to stay safe, but the MQP would have been further along if completed under normal circumstances. Overall, there has been an impact on the mental health among WPI students that reaches beyond this project and into every aspect of college and life.

7 Social Implications

7.1 Robot Safety

One of the primary social implications of this project is interaction with our robot. In the intended final use case, the robot should never interact with humans while it is moving. This is regulated through the disable option already in the FRC software. However, during development and testing, team members are likely to be in close proximity and even interacting with the robot. To ensure safety, a few basic guidelines were followed.

1. If not actively in use, robot motion should be disabled in software.
2. Be cautious of pinch points - don't stick your hands in the robot.
3. Warn those around you before starting an autonomous program or teleoperation.
4. Maintain an easy way to shut down power to the robot and inform everyone.
5. Be aware of the mass of the robot when lifting/moving.

The final lunar robot would be required to have an E-stop button in bright red to make shutting down the robot easy in case of emergency. In competition, breathing equipment and dust resistant suits would need to be worn when accessing the competition field because the BP-1 regolith simulant is very fine grain and can be abrasive to the lungs. If the robot makes use of a Lithium battery, that must also be handled and charged with care. For testing, the team used regular Lead-acid batteries, which are less sensitive but should still be treated with care when charging.

7.2 Implications of Lunar Mining

Although the competition itself does not actually involve sending the robot to the moon, its greater purpose as part of the Artemis Program is to enable creative problem-solving in lunar research with the end goal of returning people to the moon. This is generally regarded as a first step to further exploration or even colonization of the solar system, opening up much wider social implications. An autonomous mining robot could help astronauts by removing the need for physical labor and exposure to the inhabitable environment of the lunar surface. Safety protocols would need to be observed, but a mining robot would generally act independently or be remotely controlled by astronauts.

Mining materials on another celestial body is an interesting concept. While there is no life on the moon, mining would disrupt the natural landscape, and any material removed is not renewable. There is little risk of any direct danger to people as a result of lunar mining, but there is a question of if these natural resources should be tapped and eventually exhausted in the effort to colonize the solar system. The effects of any large scale operation mining for materials on the moon should be deeply examined for long-term effects. Still, the

scientific knowledge to be gained from digging a little deeper beneath the moon's surface could provide invaluable information about the early history and formation of both the moon and the earth.

8 Conclusion

This project was inspired by the NASA Robotic Mining Competition. There were four main elements of the project: design, analysis, construction, and testing. Understanding the competition and previous robotic solutions was crucial to developing a unique robot to meet the expectations of the competition. The choice of material and improvement of mechanisms in the design came from analysis of the many forces acting upon the robot. Fabricating the excavator and developing the test software allowed for testing in real digging conditions. The tests produced positive results, showing that the excavator would be able to exceed the requirements of the challenge, validating our design. As this is a reoccurring MQP, we hope future teams working on this project will be able to use and improve our design. This robot and project are just one of many reaching for the stars.

References

[1] NASA, Nasa robotic mining competition (rmc) lunabotics 2021," Available at https://www.nasa.gov/offices/education/centers/kennedy/technology/nasarmc.html (2020/04/01).

[2] Astrobotics the univerisy of alabama," https://www.alabamaastrobotics.com/ (2021/05/05).

[3] Nasa rmc 2019 (glennelope)," http://www.cwrubotix.org/nasa_robot/2019/01/01/nasa_rmc_19.html (2021/05/05).

[4] 2019 robotics mining challenge," https://www.youtube.com/watch?v=VahaU7XIT58&t=20862s (2021/05/05).

[5] N. K. Woodward, E. M. Carson, J. Rivadeneira, and P. W. Peterson, Nasa robotic mining competition," *WPI MQP Database*, p. all, 2016.

[6] R. C. Jacques, Y.-s. Wu, D. L. Bozzuto, B. G. Peterson, and A. M. Jaeger, Markhor: Robotic mining platform," *WPI MQP Database*, p. all, 2017.

[7] T. Hagen, R. A. Khan, P. Patias, K. Kumar, and N. M. Castelino, Ibex: Robotic mining platform," *WPI MQP Database*, 2018.

[8] A. H. P. Dao and L. A. Delatorre, Rove: Robotics mining platform," *WPI MQP Database*, 2019.

[9] C. S. Freedman, H. J. Burack, K. C. Bimonte, J. F. Hogan, N. L. Kuberka, and M. Hogan, Nasa rmc: Lunabotics 2020," *WPI MQP Database*, 2020.

[10] Drivetrain losses (efficiency)," Available at https://x-engineer.org/automotive-engineering/drivetrain/transmissions/drivetrain-losses-efficiency/ (2021/05/04).

[11] Vex robotics motor data - bag motor (217-3351)," https://motors.vex.com/vexpro-motors/bag-motor (2021/05/04).