Assignment 2A Presentation
EECE 443

Team 2: MCU TNC Design

Kaleb Leon – C00094357
Kobe Keopraseuth – C00092349
David Cain – C00043561

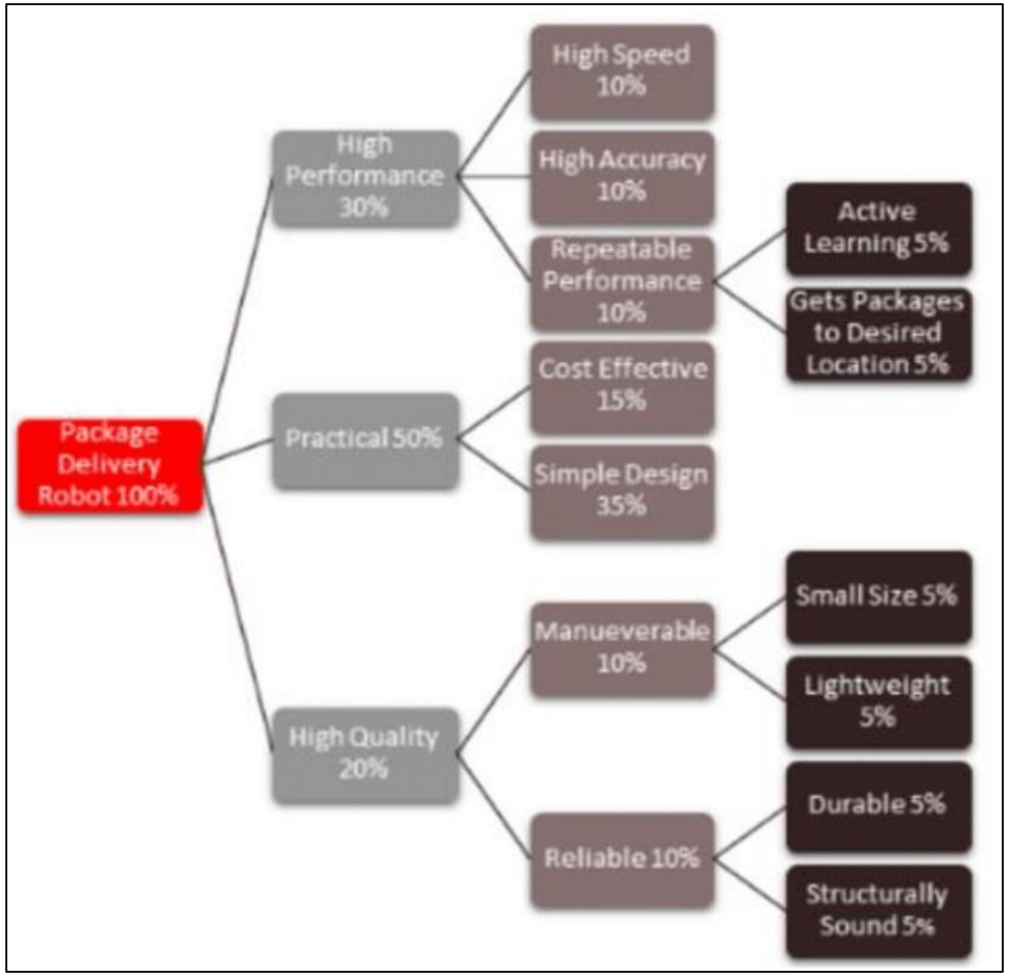
March 26th, 2020

MCU TNC Assignment 2A
Peer Review of Team 1: Package Delivery Robot

David Cain, Kobe Keopraseuth, and Kaleb Leon

Cons

- Included objective tree is blurry.
- Conclusions section is just the placeholder text.
- Some sections are just copy and paste of the same content.
- Flow Diagram is confusing to follow. Algorithm shown needs more elaboration.
- Alternatives and tradeoffs section does not go over software alternatives, microcontroller alternatives, or servo alternatives.
- Machine learning is mentioned in the first sentence of the paper but has no firm documentation in the paper.
- Schematic:
  - Bluetooth module is shown but not referenced in the paper.
  - Wiring is not neat, this makes the connections unclear.
  - In the paper, the microcontroller is stated to be powered via a voltage regulator, this is not shown.
  - In the schematic, the microcontroller is stated to be powered via USB, **conflicting with previous statement**.



Team 1’s Provided Objective Tree

V. CONCLUSIONS

The summarizing conclusion should go here. Here is where you summarize whether or not your project worked according to its requirements and the importance of your project and its

Team 1’s Provided Conclusion Section

Similar Content On Separate Pages of the Paper

Alternatives and Tradeoffs

Alternatives and Tradeoffs Considerations

A major consideration with this project is the method of which the robot will pick up blocks and store them. At the moment we plan on having an arm that can pick up the blocks and hold onto them until it will need to drop them off. If later on in the design process we need to change this plan, we have other proposed ideas of accomplishing this task. We have considered making a more advanced version of the arm if we are able to finish tasks ahead of schedule. This design would consist of a storage platform that the arm can place blocks on for later delivery. If our budget does not allow for the arms to be utilized in this build, we have also considered the option of using a type of scoop that can be placed under the robot. This scoop would work similar to a forklift, as a device would slide under the block, lifting it off the ground and placing it in its designated drop off location.

We also have different designs planned on how the robot will traverse the board. Our current planned option is to have a specific color of tape placed around the road that the robot will drive down. It will have a sensor in the front/sides that will be able to keep the robot on track. A second option we have considered is to utilize machine learning in order for the robot to learn about its environment. This would be an achievable plan if other aspects of the robot are finished in time based on our time feasibility analysis.

From Page 5

Alternatives and Tradeoffs Considerations

A major consideration with this project is the method of which the robot will pick up blocks and store them. At the moment we plan on having an arm that can pick up the blocks and hold onto them until it will need to drop them off. If later on in the design process we need to change this plan, we have other proposed ideas of accomplishing this task. We have considered making a more advanced version of the arm if we are able to finish tasks ahead of schedule. This design would consist of a storage platform that the arm can place blocks on for later delivery. If our budget does not allow for the arms to be utilized in this build, we have also considered the option of using a type of scoop that can be placed under the robot. This scoop would work similar to a forklift, as a device would slide under the block, lifting it off the ground and placing it in its designated drop off location.

We also have different designs planned on how the robot will traverse the board. Our current planned option is to have a specific color of tape placed around the road that the robot will drive down. It will have a sensor in the front/sides that will be able to keep the robot on track. A second option we have considered is to utilize machine learning in order for the robot to learn about its environment. This would be an achievable plan if other aspects of the robot are finished in time based on our time feasibility analysis.

From Page 11

Experiments, Testing, and Data Results

IV. Experiments, Testing, and Data Results

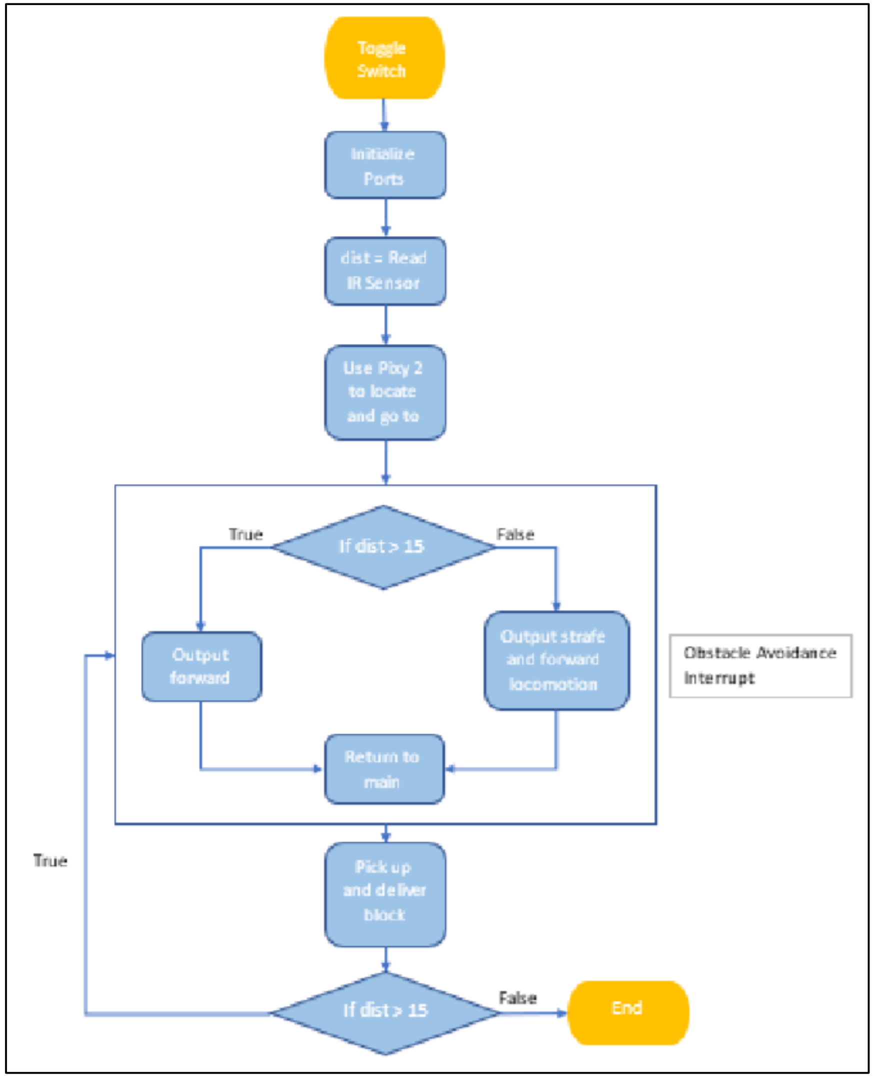
Our experimentation so far has focused on how to optimize the movement of the robot. Our initial design that was prebuilt for us had multiple issues that we discovered through testing. By attempting to drive the robot in a perfectly straight line, we found that the robot drifted about one foot to the right for every five feet it traveled forward. In addition, the horizontal movement of the robot was not functioning at all. By creating a new frame for the motors and wheels to be mounted on, we were able to see the omnidirectional wheels functioning, but not well. At this point, we found that we needed to adjust the voltage levels to the motors in order to achieve a sufficient amount of torque to drive the wheels in the horizontal direction. After many different tests with different voltages, we found that about 8 volts to the motor is optimal. Any higher voltage and the wheels would lose traction with the ground, and any lower voltage will not supply enough torque. After these experiments and tests, we were able to almost completely eliminate drift by the robot and we now have the capability of horizontal movement.

From Page 5

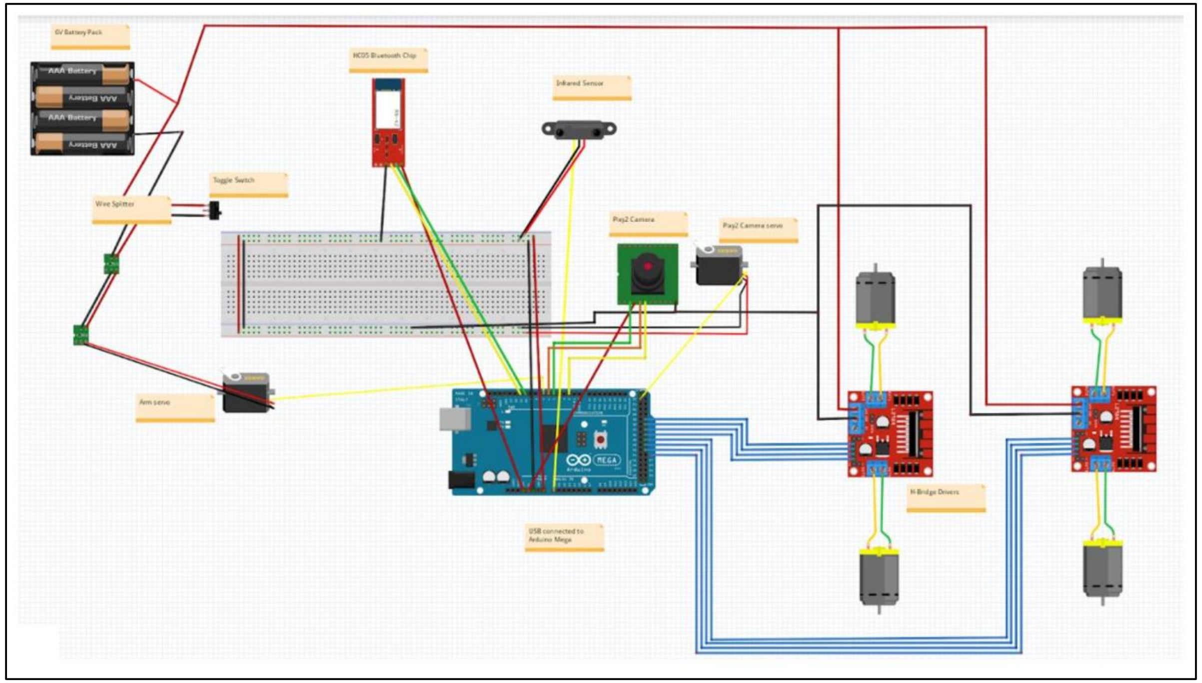
Experiments, Testing, and Data Results

Our experimentation so far has focused on how to optimize the movement of the robot. Our initial design that was prebuilt for us had multiple issues that we discovered through testing. By attempting to drive the robot in a perfectly straight line, we found that the robot drifted about one foot to the right for every five feet it traveled forward. In addition, the horizontal movement of the robot was not functioning at all. By creating a new frame for the motors and wheels to be mounted on, we were able to see the omnidirectional wheels functioning, but not well. At this point, we found that we needed to adjust the voltage levels to the motors in order to achieve a sufficient amount of torque to drive the wheels in the horizontal direction. After many different tests with different voltages, we found that about 8 volts to the motor is optimal. Any higher voltage and the wheels would lose traction with the ground, and any lower voltage will not supply enough torque. After these experiments and tests, we were able to almost completely eliminate drift by the robot and we now have the capability of horizontal movement.

From Page 11



Team 1’s Provided Flow Chart



Team 1’s Provided Schematic

Pros

- Included table of contents is helpful for traversing the paper.
- Paper is well organized to the scholarly format.
- Project analysis is well organized and explains each aspect thoroughly.
- Previous comments by our team are shown to be included in this version of the paper, most notably in the alternatives and tradeoffs section.
- Experiments, Testing and Data Results section is informative about how the team is planning to test and evaluate the performance of the project.
- FMEA table is thorough with problems and solutions.
- Level 1 diagram is detailed.

FMEA (Failure Modes and Effects Analysis)			
Function	**Failure Mode: What could go wrong?**	**Failure Causes**	**Actions to Reduce Occurrence of Failure**
Mecanum wheel	Wheels could become loose and possibly fall off. Robot could stop moving. Wheels could move at different speeds.	Poor connection to gear motor. Loss of power from arduino to wheels. Disconnected wire(s). Programming errors.	Ensure connection to gear motor is secure. Ensure batteries are sufficiently charged. Connect wires that avoids all possible ways of possibly disconnecting.
Navigating the board	Robot could bump into the perimeter of the game board and game pieces such as "pedestrians" and "buildings". Robot cannot stay on the road.	Failure of perimeter sensing sensors. Coding errors. Poor commuincation between sensors, arduino, and wheels. Failure of line following sensors.	Review any coding errors coming from perimeter sensing sensors and correct them.
Image Processing	Robot is not detecting colors. Robot cannot differentiate between colors from the buildings and blocks.	Pixie 2.0 Camera coding failure and/or Pixie 2.0 Camera is not functioning properly	Throughly test image processing at every stage during the coding stage.
Obstacle Avoidance	IR sensor failure. Lack of communication between IR sensor and arduino.Code not functioning properly.	IR sensor not functioning properly. Coding errors.	Analyze IR sensors and code before every test. Purchase new IR sensors if defect or continual failure occurs.
Finding blocks	Pixie 2.0 Camera mistakes block for another game piece. IR sensor detects block and confuses it for an obstacle thus not allowing the Pixie 2.0 camera to "read" the block.	Communication between IR sensor and Pixie 2.0 camera not functioning. Failure of IR sensor and/or Pixie 2.0 camera components. Not enough light for the Pixie 2.0 camera to properly "see".	Ensure wiring of IR sensors and Pixie 2.0 camera is correct. Check for coding errors. Make sure the room is well lit for the Pixie 2 Camera to "see". Replace IR sensors if needed.
Picking up/delivering blocks from correct destination	Robotic arm could not firmly hold the block thus it loosening from its grip. Robotic arm lacks precision and could not precisely deliver blocks in designated area. Robot could knock building downs. Robotic arm could interfere with objects on board.	Arm joints or clasps are not tight enough to bear the weight of the blocks. The joints of the arms are too loose causing flailing. Arm is too big and/or cannot remain flush with the robot thus adding more width to it.	Modify the arm to ensure that the joints and clasps can bear the weight of the blocks. Make sure that the arm will retract back to robot when not in use so that it will not interfere with anything on the game board.
Game Board	Roads are too small for the robot to travel on. Buildings are not adequately spaced apart.	Error of the game board designer. Robot pushed buildings too close and/or too far from eachother.	Correctly measure the width of the robot to ensure that the roads are wide enough for the robot to safely move around without touching any game board objects.

Team 1’s Provided FMEA Table