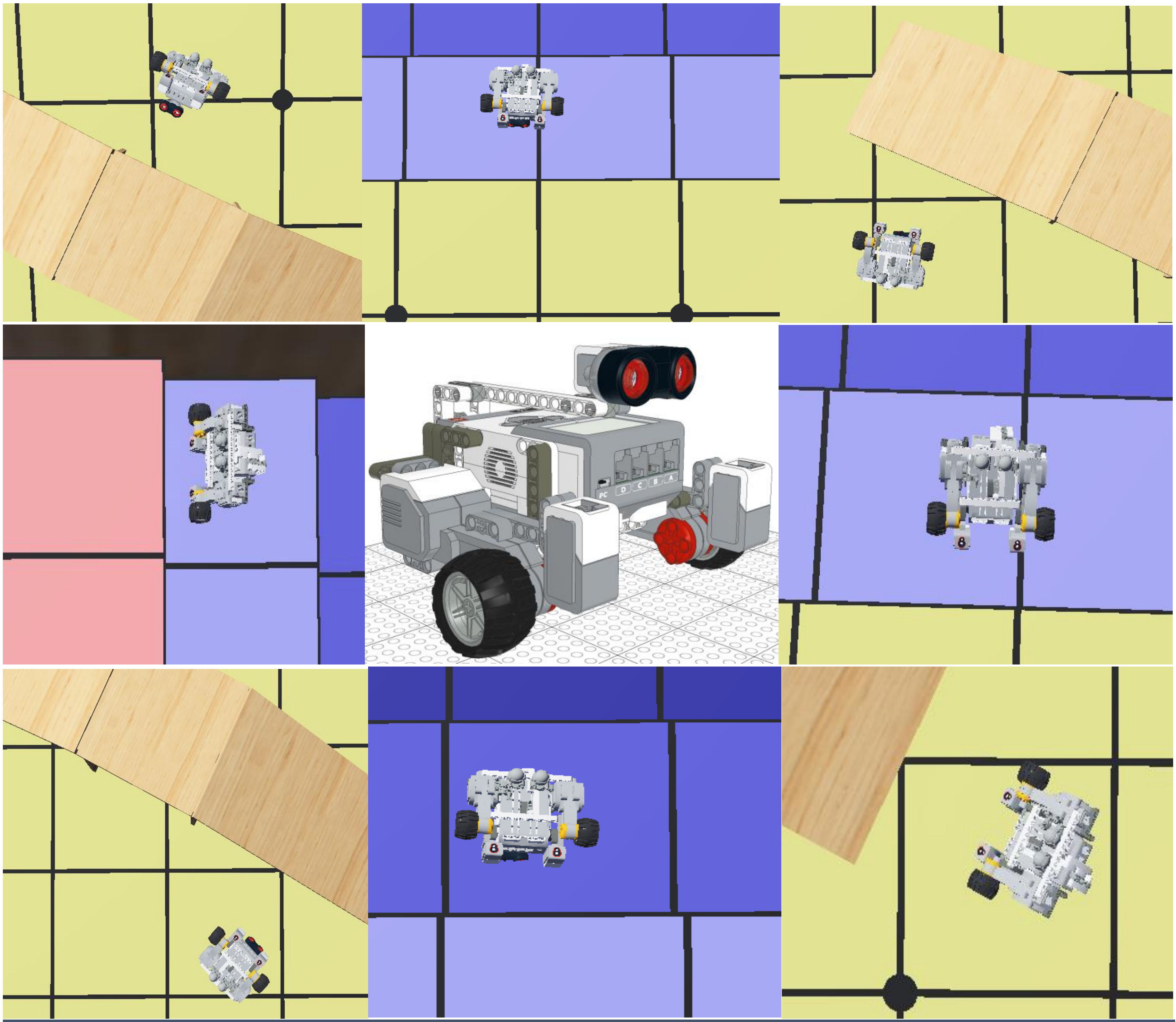

Team 1 RobotExpress

Requirements Document

Yutong Wang, Shichang Zhang, Junjian Chen, Linpei Duan, Lide Cui, Dominic Chan ECSE 211

THE REQUIREMENTS DOCUMENT

Project: Obstacle track-racer – A competitive robot design project

Task: Design and implement a working model out of EV3 Lego sets that would autonomously navigate to a racetrack on an island shown in the world map and complete as many laps as possible within the 5-minute time limit, eventually returning to its starting point.

Document Version Number: 2.0
Date: 27/03/2021
Individuals Responsible for the Document: Angelina Duan, Dominic Chan

Edit History:

Version 1.0: Angelina Duan, Dominic Chan - 06/03/2021 (Completing the initial version of the document)

Version 2.0: Dominic Chan - 27/03/2021 (Updated the document, added constraints/limitations, capabilities, and resolved unknowns)

1.0 TABLE OF CONTENTS

2.0 CAPABILITIES

2.1 PURPOSE

For this design project, we are required to implement a competitive-based game between two robots. To start off, each robot would be placed in a corner, either in the green zone or the red zone. The robot in each corner would be in their own respective team. Here, both robots would start off facing a random orientation, awaiting for the game parameters to be downloaded from the game server through a provided WiFi Class. These downloaded parameters describe the entire layout of the playing field, which includes information such as the position of the bridges between the colored and main island, the island dimensions, and the waypoints, etc. With this information provided, the robots must then localize themselves to face the front, so that they can move forward and cross the interconnecting bridge in order to reach the main island.

At the main island, there's a racecourse that is defined by a set of 10 waypoints provided by the downloaded game parameters. Each robot seeks to find the nearest waypoint to determine the direction it has to move to get to the next waypoint, where the sequence of waypoints forms a path that follows the racetrack for which each robot travels. The goal of each robot is to be able to avoid collisions with one another as well as the obstacles around them, completing as many laps as possible on the racetrack within the 5-minute time limit. In addition to this, the robots will be required to return to their respective starting points (by the corners of their starting islands) before the 5 minutes are up.

Note that every time a specific task is done by each robot, it shall emit a sequence of "beeps": 3 beeps once the localization process is completed, 3 after the robot proceeds to the nearest waypoint, and 5 beeps once the robot has successfully returned to its starting corner.

2.2 SCOPE

For this project to be successful, we expect our final product to be able to complete the maximum number of possible laps within 5 minutes, whilst avoiding obstacles and collisions along the way. To reach this goal, we will be repeatedly designing, constructing, coding, and testing our robot built from the Lego Mindstorms EV3 Kit, improving upon each design and refining it so that it can complete its tasks to the best of its abilities.

Our final product must be capable of the following:

- The robot must be able to receive information through WiFi. (Client need)
- The robot software must be able to make use of the game parameters. (Implied need)
- The robot must be able to localize itself. (Client need)
- The robot must be able to cross the bridge successfully. (Implied need)
- The robot must be able to detect the waypoints. (Client need)
- The robot must be able to detect and avoid the obstacles and other robots. (Implied need)
- The robot must be able to return to its starting corner. (Client need)

- The robot must be able to complete as many laps as possible within the time limit. (Client need)
- The robot must be able to indicate its operational status to the user through a sequence of beeps. (Client need)
- The robot must be able to move around the racetrack with wheels. (Implied need)

The above capabilities also set limitations to our robot's performance and design:

- The speed of the robot is limited by its set time.
    - The performance of each robot with regards to completing everything is determined by how fast it does its job. For example, if it's too slow, it won't be able to complete everything within 5 minutes.
- The Robot size and thus mobility are limited by the size of the bridge.
    - As the robot has to cross the bridge to the main island, it cannot physically be larger than the size of the bridge, or else it won't be able to cross it. This influences the robot's physical mobility; the ability to maneuver around the track and obstacles.
- The path of the robot is limited by the set waypoints and game parameters.
    - The path the robot takes depends on the layout of the playing field, as well as the positions of different obstacles, impacting its performance.
- The mechanical design of our robot is limited by the need for mobility and stability.
    - The hardware design of the robot must be stable and mobile for it to successfully cross the overpass and avoid obstacles.
- Components are limited to those available in the Lego Mindstorms EV3 Kit.
    - The performance of the system is constrained entirely upon the maximum effectiveness of each component of the Kit.

Unclear/Grey areas regarding the specifics of the project:

1. The given specification only shows robots starting from corners 2 and 3, can they instead start in corners 1 and 3?
    - Yes. The layout of the playing field is not fixed, meaning that each player may start in any of the 4 corners. The playing field needs to be changed accordingly based on the game parameters. In this case, corner 1 will be made either red or green, and a bridge will be constructed from that corner to the main island instead.
2. Would the rest of the playing field change according to which corners the robots start? If so, how would it change?
    - Yes, the entire layout would change according to the game parameters. There is no fixed way in which the layout would change based on where the robots start.
3. Would the obstacles be scattered everywhere around the playing field or only on the racetrack?
    - The obstacles will only be scattered around/on the race track.

2.3 CONSTRAINTS

The constraints of our design come from those outlined in section 2.2 (Scope), the hardware/tools that are available for use, the budget of the project, and the technical skills of our team members. Thus, in addition to the constraints and limitations outlined in section 2.2 (Scope), we are also given general constraints from the client:

- Hardware: Our robot must be built from the components of the 3 Lego Mindstorms EV3 kits. It can implement at most 4 sensors and at most 4 motors.
- Software: Our software must be able to run on an ARM 9 Processor at 300MHz, with 16 MB of Flash memory and 64 MB of RAM.
- Resources: We must use the Webots 3D robot simulator to simulate the behavior of the robot in the playing field. The capability of each team member is outlined in the *Capabilities Document*.
- Environment: The environment of the playing field and robot is constrained by the Webots 3D robot simulator.
- Budget: The budget constraints are outlined in section 2.4.

2.4 BUDGET

The budget for this project includes the costs of personnel and components that may need to be acquired.

For our personnel budget, our total budget must not exceed 351 hours as a team or 58.5 hours per team member, which equates to approximately 9.75 hours per week per team member. So, by spending 9 hours a week per person on this project, with 6 weeks to complete the project and 6 members on our team, we expect our whole project to be completed in $9 \times 6 \times 6 = 324$ hours.

On the other hand, the Lego components and the LeoCAD, Eclipse, and Webots software used to construct and simulate our robot for this project are free so there would be no additional costs to our budget.

2.5 USER FUNCTIONS

In general, although the user can interact with the device and receive inputs through the use of a key or buttons since the robot's WiFi interface allows for Webots to directly transfer information to the robot, the user would instead interact with the device mainly through the use of the Webots programming interface.

Thus, before the operation, the robot's WiFi interface would allow game parameters to be downloaded to the robot. However, during operation, we cannot interact with the robot, and we do not need to. Since the robot is fully autonomous, where the robot is set up and run in a batch processing system, this allows for the robot's tasks and processes to be run sequentially without end-user interaction. The robot would only stop after all the processes have elapsed within the 5-minute time constraint, which is the end of the competition.

Therefore, we can only interact with the device before the operation and after the operation.

2.6 OPERATING ENVIRONMENT

Due to the ongoing Covid-19 pandemic during the completion of this project, this project would hence be completed through virtual means, though its application is based on real-life (and so should work in real life). Therefore, the playing field is a designed world platform in Webots measuring 15’ by 9’, and the composition of the competition surface can be assumed to be of low surface friction across all of the starting field, bridges, sea, main island, and racetrack. This could influence our navigation and localization accuracy since higher surface friction could increase the traction between the wheel and the robot, reducing the magnitude of errors from wheel slips. Thus, by assuming low surface friction, if our processes function accurately in these adverse scenarios, it would then also be accurate in more ideal situations. Similarly, with regard to the ambient lighting, external noise, and general temperature of the environment, although these constraints are relatively ambiguous in Webots, it is not the case in real life and so we would have to design our robot in such a way that would allow the robot to function correctly in a multitude of situations.

2.7 PERFORMANCE

At the very least, the robot must in general be able to complete all the required tasks in the competition and return back to the starting point without failure (refer to section 2.2: Scope). As for the response time, since our robots are fully autonomous, the response time to a command is irrelevant (refer to section 2.5: User Functions). However, since the competition has a 5-minute time limit (refer to section 2.3: Constraints), the robots do need to be relatively quick and adept in determining the position of the next waypoints and navigating through them while avoiding collisions.

3.0 LIST OF UNKNOWNS

3.1 UNKNOWNS AND QUESTIONS

All initial unknowns and questions have been resolved.

4.0 COMPATIBILITY

4.1 COMPONENT RE-USE

Since the client did not decide on the need for new and unique designs, components, and software, we can hence reuse anything from previous laboratories to help us with this design project. In particular, the Odometer, LightLocaliser, and Ultrasonic Localiser classes in Lab 5, as well as the testDriver class methods from Lab 2 and Lab 3 would prove to be useful in coding the navigation, obstacle avoidance, and localization of the robot in this project. This code can simply be reused by changing it so that it fits the needs of this project. On the other hand, while the hardware and playing field should be redesigned from scratch to best fit the needs for this project, we can still re-use the knowledge learned from previous laboratories to implement this.

4.2 COMPATIBILITY WITH THIRD PART PRODUCTS

The client only permitted the use of products and components from Lego, advocating it to be constructed on the LeoCAD software, before simulating it on Webots. The only third-party product used is the WiFi class provided by the client which is used to transfer the game parameters to the robot.

5.0 GLOSSARY OF TERMS

Flash memory: An electronic non-volatile computer memory storage medium that can be electrically erased and reprogrammed.
Batch processing: The running of jobs and tasks without the need for end-user interaction.
Budget: The cost of the project in terms of the time spent by each member on the project.