RoboCup Junior Australia

Open Rescue 2014

TEAM WUKONG:

Alexander Pham

Oliver Locke

Contents

Problem Definition:

The Scenario – "A terrible earthquake has hit the city and caused a large chemical storage unit to rupture spilling thousands of litres of toxic chemicals in the centre of the city. There is a person trapped in a sinking rescue capsule (the victim) in the middle of the chemical spill. Rescue crews are having trouble entering the city with the amount of rubble around, and rescue from the air has also been ruled out due to the noxious gases rising from the toxic chemicals directly above the spill. It has been decided that the best form of rescue is the deployment of an autonomous robot that can navigate to the scene, rescue the victim and exit the chemical spill.

The robot can be deployed at the start tile (City Limits) or air dropped to any tile that is at least two (2) tile away from the chemical spill (Drop Zone)."

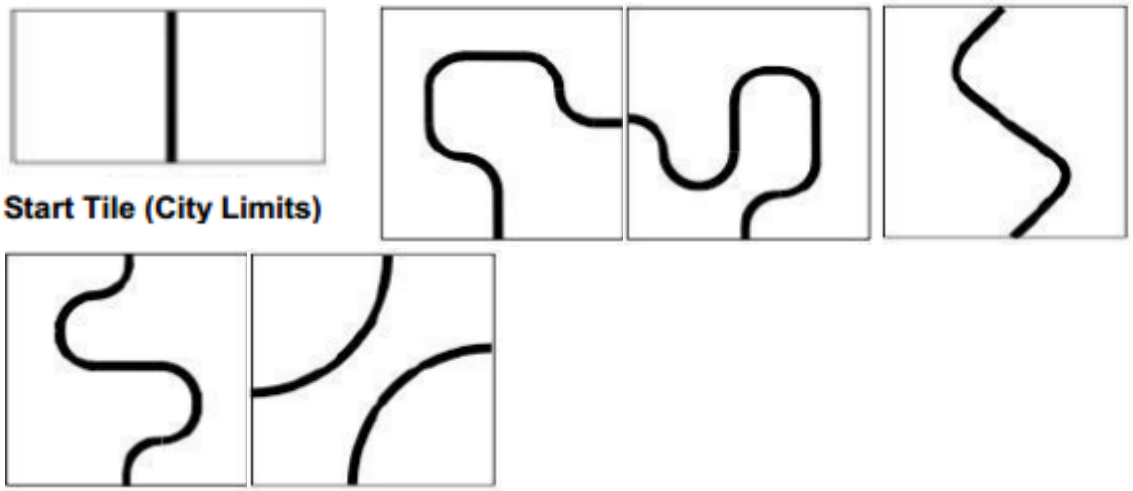
Open Rescue: "The robot must navigate to the scene and remove the rescue capsule from the chemical spill and place it in its original orientation safely on the evacuation platform for later collection by an aircrew. The robot must then save itself by exiting the chemical spill via the 'Spill Access Point'"

The Problem Decomposed – we must create a robot that is able to:

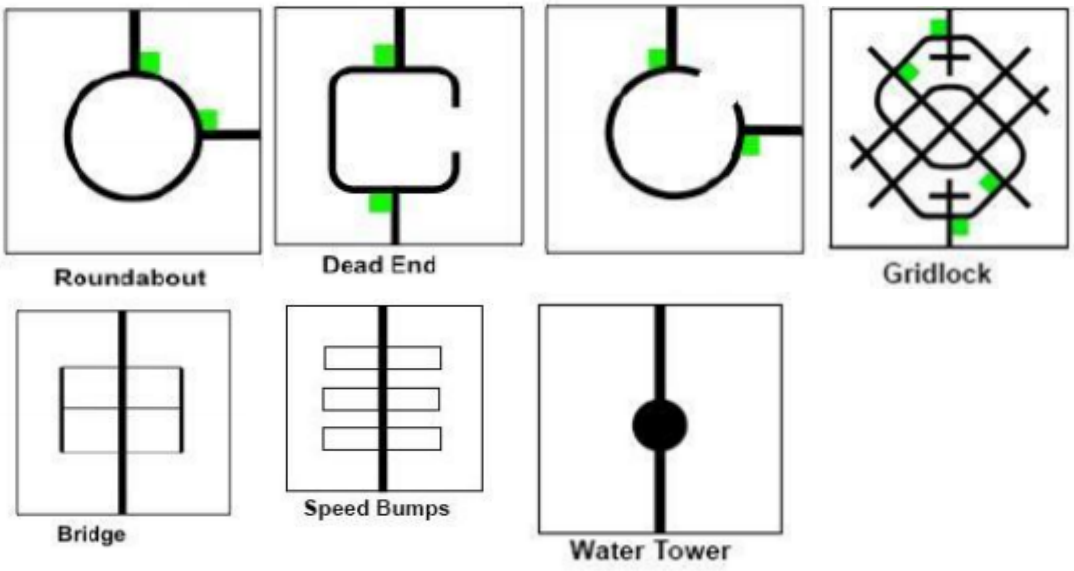
- Follow a course via a path
- Reach the end of the course and then detect a can(rescue capsule) and place it on a block/evacuation platform
- After rescuing the capsule, the robot must exit the final tile

The course – The course consists of:

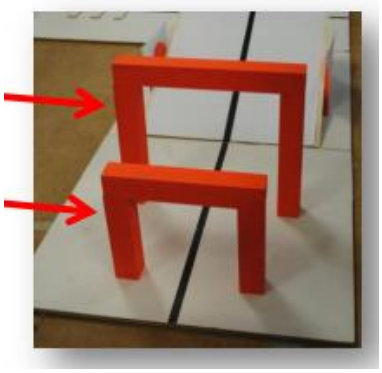
Basic tiles like the following where the robot must follow the black line:

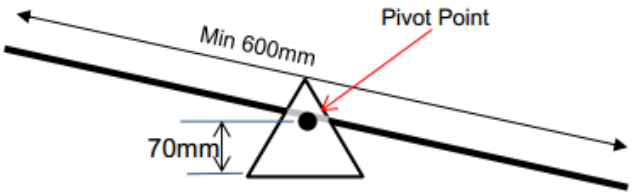


Harder tiles where the robot must successfully overcome an obstacle, or follow the green square while still following the line:

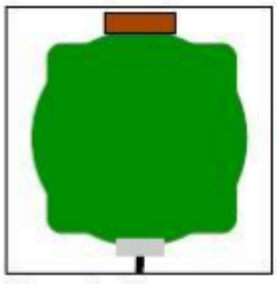


Obstacles that the robot must surpass:





End Tile where the robot will detect the rescue capsule and place it on the block:



Planning:

Roles of Team Members

Alexander – Lead Programmer, builder

Alex will solely be developing the code for the robot up until the point where the robot must pick up the can. He will also assist Oliver in building some parts of the robot as well as designing the robot.

Oliver – Lead Builder, programmer

Oliver is the core builder for our robot and will execute design plans as well as help out in the coding for the rescuing of the can/capsule.

Problem Solutions

Basic Tiles: To follow the black line, the robot must be able to detect, using the light sensors, the difference in colours on the tile. In this case, in order for the robot to "detect" the black line, we will constantly make the robot test to see if it's on white and turn accordingly.

Harder Tiles: To follow the green square, the robot has to detect a change reflection of light using the light sensor and turn towards the green square.

Obstacles: If the touch sensor is hit by the water tower, it must navigate around the tower.

End Tile: The robot must detect the distance from the can using the distance sensor and then travel towards it, pick it up using our mechanism and then place it on the block and reverse out of the tile.

All sensors seem to be available at school.

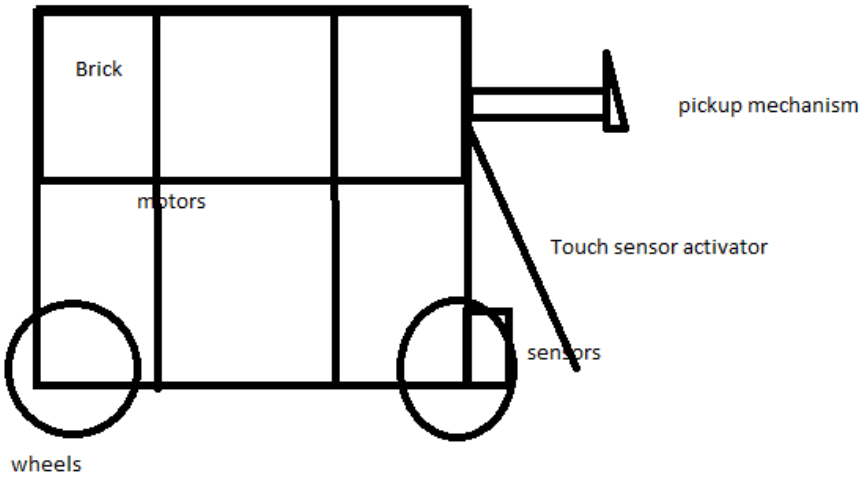
Task Planning

We will spend the first 10 weeks constructing our robot, making sure that the end result is stable and fits size requirements. The rest of our time will include programming, debugging, testing and preparing our robot for the competition.

Debugging and testing will only take up a minor portion of our time (1-3 weeks).

Solution Design:

The design of our robot will hopefully look similar to the following:



Implementation:

Week 2, Term 1

The hardware and software we decided to use was RobotC and Lego Mindstorms because it was readily available at our school.

Week 5, Term 1

Since we (the members) had worked together previously, we knew what roles we suited best and so we allocated it to each other. We started drafting up potential ideas and designs for our robot.

Week 8, Term 1

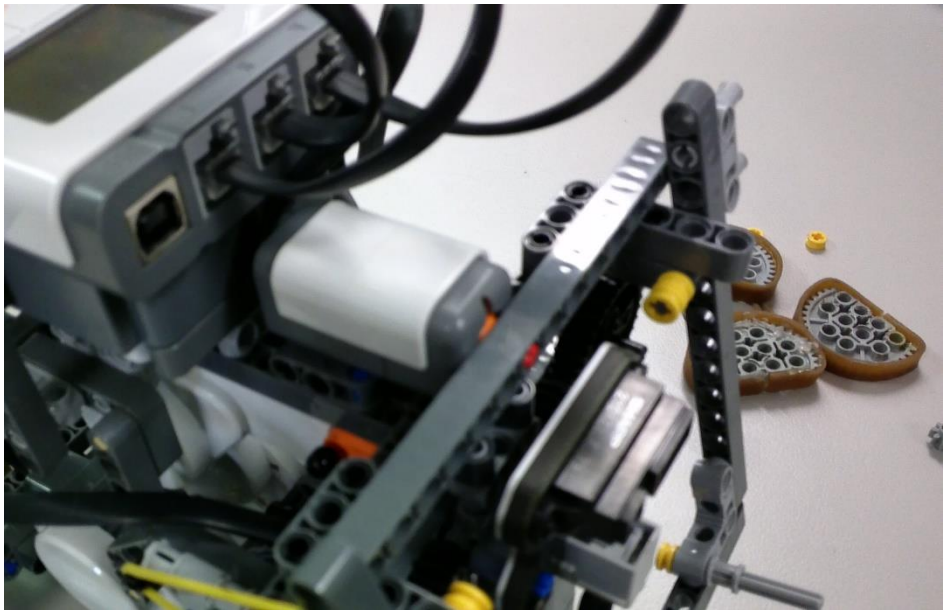
We began to build our robot, starting with the structure of the robot as well as the placement of the NXT Brick. After that, building the rest of the robot was simple except for the touch sensor and distance sensor.

Week 2, Term 2

We attached the touch sensor onto the robot at the front and began to work on the can pick-up mechanism. Alex began setting up the motors and sensors on RobotC.

Week 3, Term 2

We decided to invert the touch sensor as it was easier to activate using the water tower in this form, whereas it would push the tower over if the touch sensor was not inverted



Week 5, Term 2

Construction of the robot was basically complete and so the coding could begin. Alex managed to get the robot following the black line with ease and adjusted speeds to

try and improve the time it took for the robot to get past the course. We used this code:

Code hidden, no copying!

Week 8, Term 2

The pick-up mechanism was complete so we attached it to the body of the robot. Our code was also moving along and Alex was able to make the robot follow the green

Code hidden, no copying!

square and navigate slowly around the water tower. Our code currently looks like this (setup of sensors and motors are left out):

Week 1, Term 3

The only tile left to be conquered other than the end tile was the gridlock. We managed to overcome it by entering this into our code:

```
else if(RS > lowblack && RS < highblack && LS > lowblack && LS < highblack)//gridlock,both black
{
  motor[left] = 30;
  motor[right] = 30;
}
```

Week 2, Term 3

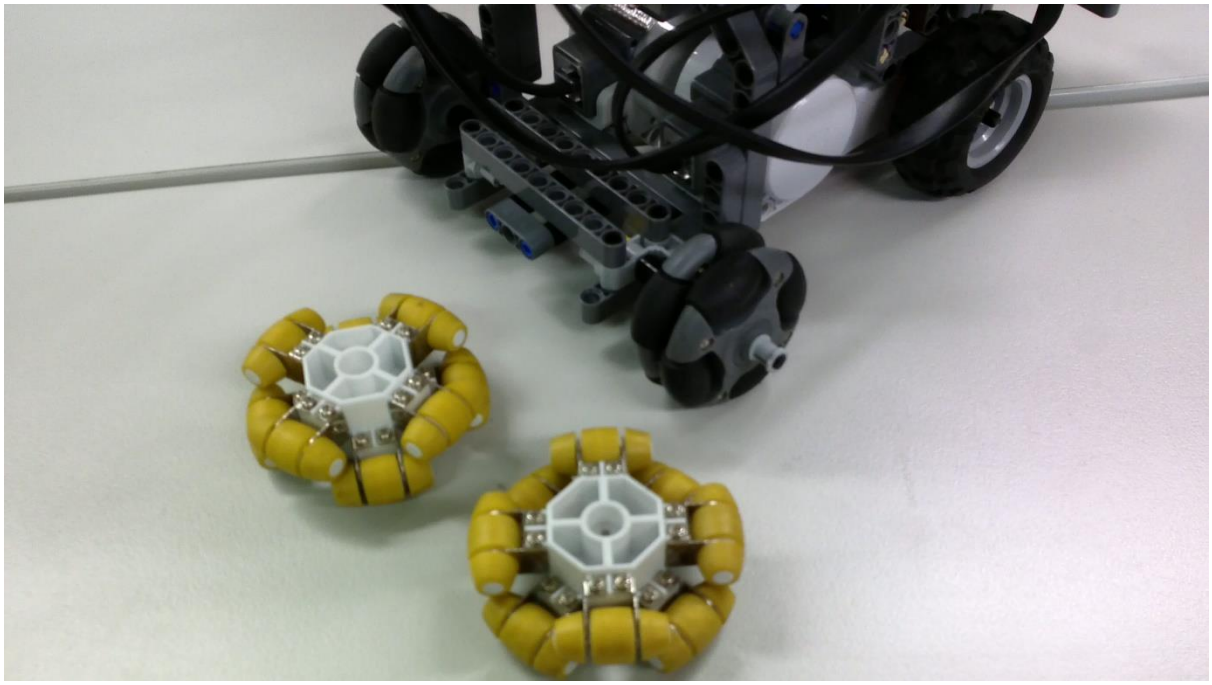
Tweaking to the code was done to exhibit more accuracy. Our robot can now follow the whole course almost 100% of the time, excluding the picking up of the can.

Week 3, Term 3

Oliver began to help out with the final coding of the robot. He worked out the maximum distance in which the distance sensor could detect objects and made the robot move towards the direction if it detected anything.

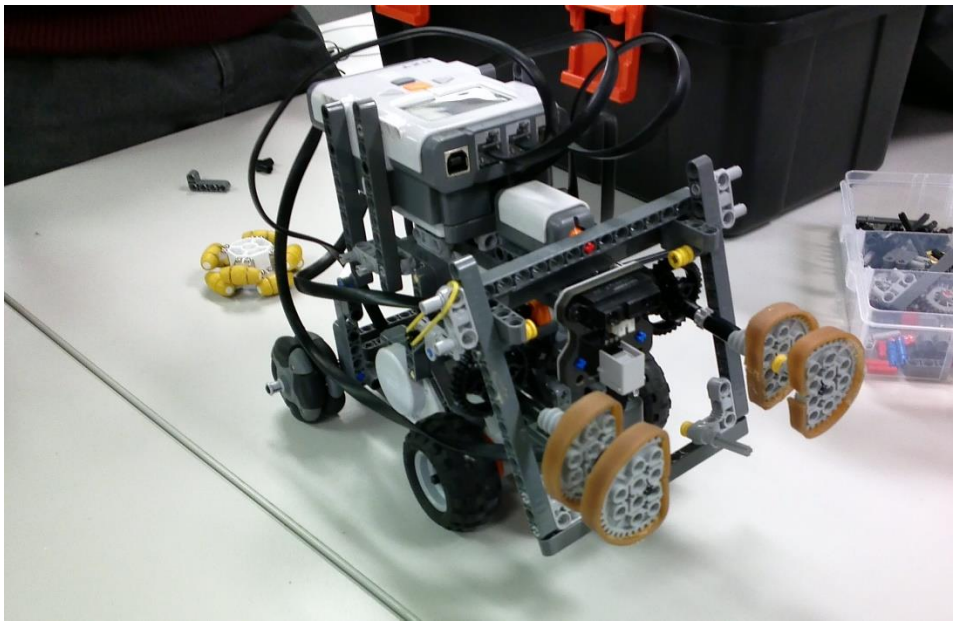
Week 4, Term 3

We changed our back wheels into smaller ones to make sure the robot would not fall off tiles and we added weights to the centre of the robot. Our code is basically finished and just requires debugging.



Evaluation:

The following is what our robot currently looks like –



We are happy with what we have created and do not plan on changing many key features, bar the picking up mechanism. We are quite sure that our robots comply with the rules of the competition and there is still time to edit our robot if it does not. The robot design was clear throughout the whole process and we did not make any drastic changes to our initial design.

We hope that we will be able to beat our previous attempt and place first in this year's tournament.