BALLISTA - EIN TRAUM VON ALEXANDER

Preetdeepan Prashantkumar Pradhan, Electrical Engineering & Information Technology
Otto-von-Guericke-University of Magdeburg

Abstract — In this modern era of work segregation, autonomous robots have challenged the innate perspective of mankind. They not only save human labour, but also provide assistance where no mankind has ever stepped its feet. The modification of this medieval weapon into a functioning prototype, with several uses in the field of constructions or recreational sport. The project uses the Boolean Logic partially, as the robot moves around, detects hinderances and turns 90° left.
The apparatus building requirements were from the 'Lego Mindstorms Education Set', that developed the corresponding programme in MATLAB. In order to provide a stable communication between MATLAB and the LEGO-NXT Module, the use of MATLAB Package from RWTH-Aachen was very essential.
As a result, we received a working model and a few more comments based on our development process that should be met in order to be able to use the plowing machine in practice. On the one hand there are additional sensors to detect pedestrians or other things that are not obstacles and on the other hand it would be a satellite transmitter to detect which surface has to be processed and where, for example, the earth can be taken.
Keywords — Autonomous Driving, Boolean Logic, Lego Mindstorms, Matlab.

I. INTRODUCTION

The quick uprising of robots, that outperform humans have sustainably replaced and made our lives relatively easier to work with. The use of robots, that can be easily programmed and worked with have significantly increased. The best example for the same is the Automobile sector, Packaging system and most importantly in the Food & Drugs division.

Robots have made our daily work habits much convenient, as they take over simple or maybe complicated tasks, freeing mankind to explore the unexplored, and be busy with the same.

There have been technological discoveries, inventions and most importantly discussions about improvement of existing technologies throughout the existence of humanity.

One such prime example, is the Catapult, a medieval weapon, that can be used in various other fields, especially with constructions of temporary bridges using boulders for the military in places, where human involvement is very shrewd, crude and difficult. Here is the good use of the robot, as it will be able to provide assistance.

Exactly what the project "Ballista" sets, that has been brought to life during the period of 11.02.2019 till 22.02.2019 in the LEGO Mindstorms seminar. The aim was to create a prototype as a model based on the designs of the ancient Greek Philosopher Diodorus Siculus. This is a simple illustration of what followed thereafter, the rise and fall of kingdoms, dynasties, eras and most certainly empires.

This model should, while following simple codes that can be easily brought about by any technician can cause fatal damage to the opponent with a reaction time so slow, that the model can detect obstacles, thereafter change its direction of movement and recalibrate to function again. The model follows a line (as a simplified road) recognise obstacles and test whether its projectile is affected by the

II. PREVIEW

Boolean Logic

The use of Boolean Logic is extremely essential for the robot's movement, as the detection of an obstacle greater than the robot's height would make the robot change its path of movement. The explicit and exact data is delivered using the Boolean logic, as it provides only and only two values, either 0 or 1, where specific commands were allocated to both the values.

Basic Programming

For basic programming, command overview, where codes could be written, was used. In addition to the communication between MATLAB and the NXT module, it also describes the basic program sequences, as well as the integration and use of functions. A measure had to be found to output the currently running processes.

DOI: 10.24352/UB.OVGU-2019-045

III. BODY

Figure 1. Segway

A. Brainstorming

The basic concept was a vehicle that has simulated fixed steering to make manoeuvring easier and making the tropics almost negligible. The initial point of framework was a Segway. Due to faulty measurements and imbalanced structure, the idea was scrapped. The key points of failure were carefully analysed, discussed and thereafter improved, leading the main point of focus to make sure that the vehicle was compact, robust and outstandingly constructed so that a low centre of gravity was achieved. Thus making it easier to move with objects or the armament, and reduce the risk of the vehicle tipping over and being damaged. This shall definitely not be the case with this particular model, but in practice it shall be a very important point that cannot be neglected, if the robot has to drive over rough terrain.

Figure 1. is the model representation of a Segway, and the origin of the idea of a catapult.

The initial construction of the Segway was very useable, but after fixing the wheels to the Segway, a major error was noticed and thereafter all other challenges were brought about, for example the stability due to very high and displaced centre of gravity, weight of the NXT Motor, the slow processing of the NXT Program, and the very high latency. This was the basic idea on which the structure and the project development was done. Although a beta, this seemed to be a working model, that could be improvised on and could have been made more interesting and useful.

Also, necessity is the mother of invention, which has been the guiding factor in the complete changeover of a basic Segway model into one of the greatest and the deadliest weapon of the medieval and ancient Greek era.

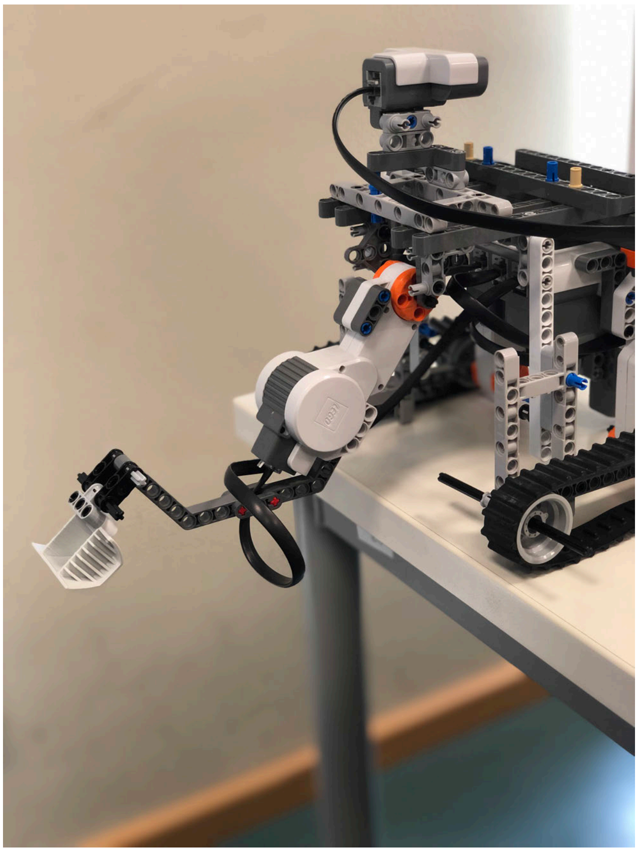

Figure 2. Head of Segway converted into Catapult head.

Figure 2. is the step one of the changeover of the Segway. The initial framework of the Segway had a head, which was later converted into the arm of the catapult, there was a programme written for the same, to assign engine force to the arm, the power supplied to the same was of 100 percent.

The robot is able to swing its arm in a semi circle with 100 percent power, which enables the object or the armament loaded to move in a projectile. This is indeed one basic movement, compared to the conventional variants of a catapult, where realignment and reloading is vital and consumes significantly large amount of time. The reconfiguration of the programme, after adoption should definitely remove all the errors and help stabilise, and achieve the required projectile with the adjustment of the angle of the arm.

After this the robot’s linear movement with the arm placement was worked upon.

The wheels of the Segway were replaced with Caterpillar tread track, that made the linear movement stable and straight. The caterpillar tread tracks were fixed to two adjoining motors that were supplied with an engine power maximum of 30 percent, which could make the robot's trajectory linear, stable and coherent. This also provided time for the other sensory movements of the robots to detect obstacles, make the necessary change of path and avoid edges to prevent falling off course during travel time.

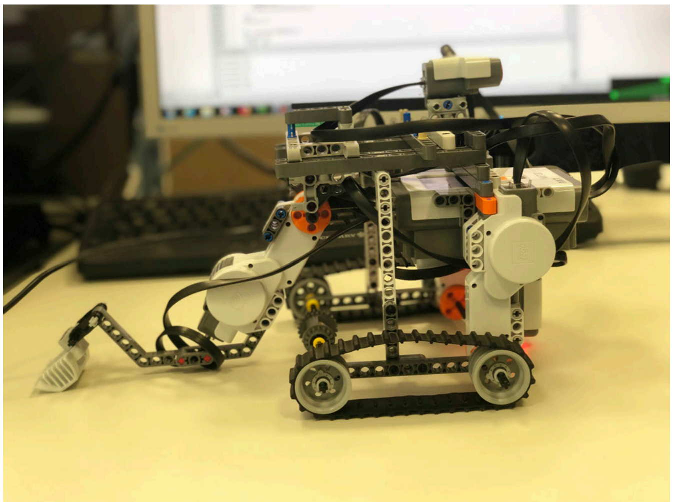

Figure 3. Final Build of Ballista

Figure 3. After the initial testing of the linear movement of the robot, the decision to install the sensors (after calibration) in order to increase the sophistication of the robot was challenging. After that it was about the robot to detect obstacles and later to test them for displacement.

Keeping in mind about the terrain change, and stability the following sensors were installed. The following NXT sensors are important for this:

- One Ultrasonic Sensor
- One Light Sensor
- One Touch Sensor

The Ultrasonic sensor is located right on the top of the robot, disguised as the head of the robot as in seen in Figure 3, and this sensor faces to the direction of travel. The former is used to detect whether an obstacle is on the road, so that the following can give a command to the adjoining engines connecting the wheels to turn, if the obstacle is greater than the height of the robot, so that any and every kind of hinderance during the projectile can be avoided. It is also used to detect when the robot can avoid the obstacle with a left turn or if it is still on the line.

As previously described, the Light sensor located in the front centre of the robot so that it can track the edges. It can measure the reflectivity of the surface, therefore if there is a hole or a gap in the road, light from the sensor will travel a longer distance hence the reflection shall be invariably low causing the sensor to realise an edge. Henceforth causing the robot to change its direction and movement. After realising an edge, the robot tends to go back a bit, and then take a left turn, since the edge could be lengthwise longer than expected by the robot.

```
OpenSwitch(SENSOR_2);
OpenUltrasonic(SENSOR_1);
OpenLight(SENSOR_3,'ACTIVE');

while 1
    % if switch is pressed he shoots with the catapult
    if GetSwitch(SENSOR_2) == 1
        handles.motorA = NXTMotor('A','Power',-100,'TachoLimit',100);
        handles.motorA.SendToNXT();
        pause(1);
        handles.motorA = NXTMotor('A','Power',10,'TachoLimit',100);
        handles.motorA.SendToNXT();
    end

    dist = GetUltrasonic(SENSOR_1);
    fprintf(['Distance:' num2str(dist) newline]);
    % drives forward until the distance is smaller than 15
    if dist < 20
        handles.motorA = NXTMotor('A','Power',-50,'TachoLimit',100);
        handles.motorA.SendToNXT();
        pause(0.3);
        handles.motorC = NXTMotor('C','Power',-50,'TachoLimit',475);
        handles.motorC.SendToNXT();
        handles.motorB = NXTMotor('B','Power',50,'TachoLimit',475);
        handles.motorB.SendToNXT();
        handles.motorB.WaitFor();
    end
    pause(0.2);

    x = GetLight(SENSOR_3);
    pause(0.1);
    if x < 600
        handles.motorC = NXTMotor('C','Power',-60,'TachoLimit',300);
        handles.motorC.SendToNXT();
        handles.motorB = NXTMotor('B','Power',-60,'TachoLimit',300);
        handles.motorB.SendToNXT();
        pause(2);
        handles.motorC = NXTMotor('C','Power',-50,'TachoLimit',475);
        handles.motorC.SendToNXT();
        handles.motorB = NXTMotor('B','Power',50,'TachoLimit',475);
        handles.motorB.SendToNXT();
    end
```

Figure 4. Essential Functions from the source code

Figure 4. portrays all the indispensable functions.

The 'while loop' function is an infinite loop and ensures the successful running of the program until user intervention manually by pressing the switch. Boolean logic plays a significant role while enabling the 'if-else' functions, as it detects the signal from the Touch Sensor as a 'condition'. Any stalling of the robot's arm can be prevented and controlled using the 'Tacholimit'. The power supplied to the same is 100 or full power which judges the projectile perfectly ensuring the desired result.

The returning power to the arm is set at 10 in the opposite direction, which ensures that the robot's arm returns to its original position easily and slowly.

The movement of the robot are endowed on two main functions. The Tacholimit was fixed on 475 after careful analysis in consideration to the rotations of the motor. The next Tacholimit was fixed on 300, it does alter any change, as the value is random and is only used to guide the Ballista to proceed in the “negative” direction and not over the edge of the surface in consideration.

The Touch sensor fixed to the robot, is actually a button that enables the user to fire the Ballista after carefully fixating the robot with the desired coordinates.

The Touch sensor is absolutely user based and has no vital connection to the movement of the robot in whole.

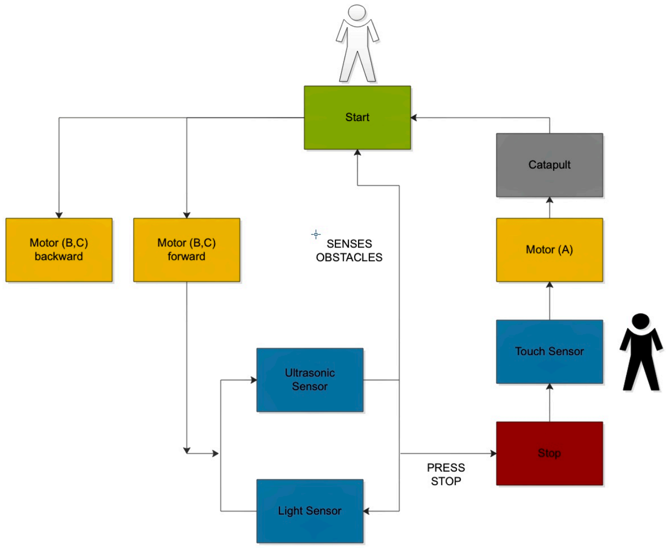

Figure 4. Program Layout
Yellow: Motors
Blue: Sensons
Start and Touch Sensor require human intervention

The basic program outlet, the basic construction and the execution has been described in the above mentioned Figure 5. The user presses the Start button, the commanding motors are driven with power and the wanted movement takes place, the light sensor and ultrasonic sensor help the robot drive and stay stable, and with the user’s command with the touch sensor enables the robot to fire i.e. move the arm and project the armament for the desired outcome.

IV. RESULT

After repetitive running, collecting, analysing and summarising the robots’, data, movement and analysis, one could conclude that the manuscript-code is usable and the original aim was achieved. The point of major improvement is the arm, a motor with a higher power efficiency can replace the already existing motor, so that the delivery of power to the arm can be sufficient for a stronger projectile.

The sensors need major improvement, technically, to avoid unfruitful errs. High precision can be derived.

V. SUMMARY

By using Boolean Logic as the basis of sensory movements of the robot, the establishment of a linear movement is possible, in spite of obstacles, gaps, holes or edges present. When the robot faces a hinderance, it finds a way independently and then continues on a new path on the marked section.

This can be considered as a basic prototype for high-grade military artillery.

For the decision-making, there should also be found a way to clearly distinguish between a disruptive factor (branches or boulders) and a normal obstacle (car or motorcycle).

VI. BIBLIOGRAPHY

I. FEIT OvGU (2019 - February) [Online]. Unterlagen zum LEGO Mindstorms Praktikum in der FEIT

II. http://www.nxtprograms.com/NXT2/segway/